university of groningen

faculty of science and engineering

Dynamic Ensemble Selection with XGBoost for Enhanced Survival Analysis of Kidney Transplant Outcomes Using NMR Metabolomics

Mortaza Akbari

university of groningen

faculty of science and engineering

University of Groningen

Dynamic Ensemble Selection with XGBoost for Enhanced Survival Analysis of Kidney Transplant Outcomes Using NMR Metabolomics

Master's Thesis

To fulfill the requirements for the degree of
Master of Science in AI
at University of Groningen under the supervision of
Dr. Marleen Schippers, Dr. Jiapan Guo. (Artificial Intelligence, University of Groningen).
External Supervisor: Dr. Tamas Szili-Török, M.D., Ömer Tarik Özyilmaz (University Medical Center Groningen, The Netherlands)

Mortaza Akbari (S4115988)

August 30, 2025

Acknowledgements

I would like to express my sincere gratitude to everyone who supported me throughout the completion of this master's thesis.

First, I would like to thank my internal supervisors, Dr. Marleen Schippers and Dr. Jiapan Guo, for their guidance and the valuable meetings we had, which helped me clarify and focus my research.

I am also grateful to my external supervisors at the University Medical Center Groningen. At the start of my project, Tamas Szili-Török provided essential guidance, and throughout the rest of the research, Ömer Tarik Özyilmaz offered continuous support, feedback, and insightful discussions that greatly enhanced this work.

Finally, I would like to thank my family, friends, and colleagues for their encouragement and support during this journey.

Abstract

Survival analysis underpins time-to-event prediction in medicine, where accurate timing matters for decisions and outcomes. In kidney transplantation, anticipating graft failure can guide monitoring, immunosuppression, and re-listing. Classical tools such as the Cox proportional hazards model struggle when relationships are nonlinear and data are high-dimensional or censored. In this context, machine learning methods, used to extract predictive signal from complex measurements, offer a pragmatic way to interpret rich sources such as Nuclear Magnetic Resonance (NMR) metabolomics.

This thesis evaluates dynamic ensemble selection (DES) for survival prediction, using XGBoost models ensembles to convert complex input data into clinically useful risk estimates. DES tailors, per patient, which models are combined - in contrast to static selection or uniform averaging - and is designed to exploit local structure in heterogeneous populations.

We study two settings. First, a high-dimensional NMR cohort of kidney transplant recipients (n=249; 30 events) serves as the primary test bed. Second, as a cross-dataset check, we apply the same pipeline to NHANES with linked mortality follow-up (lower-dimensional, larger sample). The concordance index (C index) is the primary metric. In the NMR cohort, the DES strategy that prunes and then weights models (DES-WS) outperformed static selection and uniform averaging in the external test set (peak C-index ¡0.05), while single-model selection (DES-S) was unstable. Genetic algorithm optimization of competency weights generally did not improve performance, consistent with overfitting in small validation sets. In NHANES, all strategies performed similarly (C index ¡0.81-0.82), indicating limited headroom for DES when predictors are few and the signal is more global.

These results support DES as a data-driven mechanism for interpreting complex biosignals (e.g. NMR spectra) for patient-specific survival prediction, with clear gains in heterogeneous high-dimensional settings and parity in simpler ones. Future work should emphasize larger multicenter cohorts, calibration and decision-analytic evaluation, model interpretability, and extensions to competing risks and time-varying covariates.

Contents

1 Introduction

Survival analysis is a statistical approach used to estimate the time until an event of interest occurs within a given observation period Barnwal, Cho, and Hocking (2022). It is commonly applied in medical studies, where the goal is to predict survival time or the occurrence of adverse events such as organ failure.

Traditional regression models, such as linear regression, are inadequate for complex survival analysis scenarios due to their simplistic assumptions and inability to properly handle censored or time-dependent data. To address these challenges, specialized methods such as the Cox Proportional Hazards (CPH) model Cox (1972) have been widely adopted. The CPH model assumes proportional hazards and linear covariate effects, which often fail to capture the non-linear and heterogeneous patterns present in biomedical data Pickett, Suresh, Miller, Davis, and Juarez (2021). This limitation has motivated the exploration of machine learning–based survival models that can better model complex structures.

Significant advances include the development of Random Survival Forests (RSF) and the integration of the CPH model into Gradient Boosting Machine (GBM) frameworks Ke et al. (2017); Pickett et al. (2021). These methods leverage flexible algorithms to overcome proportional hazards assumptions and effectively model nonlinear relationships, improving predictive performance in real-world applications.

Alongside these advances, ensemble learning techniques, such as eXtreme Gradient Boosting (XGBoost) T. Chen and Guestrin (2016), have emerged as state-of-the-art methods for refining prediction models. XGBoost incorporates regularization techniques to reduce overfitting and has set new performance benchmarks across regression and classification tasks. Ensemble learning methods, which combine multiple models to improve robustness, have demonstrated superior accuracy over single-model approaches Caruana, Niculescu-Mizil, Crew, and Ksikes (2004); Dietterich (2000); Zhou (2015). They typically follow three phases: generation, selection, and combination Cruz, Sabourin, and Cavalcanti (2018).

In the generation phase, an ensemble of models is created by training on a given dataset. Ensembles can be homogeneous if a single learning algorithm is used across all models, or heterogeneous if multiple learning algorithms are employed. The selection phase then chooses a single model or a subset of models to make the final prediction. This selection can be *static*, where the same subset is used for all test cases based on global validation performance, or *dynamic*, where a different subset is chosen per test instance according to its location in feature space. Dynamic selection techniques, which adaptively pick the most competent models for each case, have been shown to outperform static methods in heterogeneous settings (Ko, Sabourin, & Britto Jr, 2008).

For example, in medical diagnosis, each patient presents a unique combination of symptoms, history, and biomarker data, creating significant variability between cases. Here, dynamic selection proves advantageous, as it allows the model ensemble to adaptively choose those 'weak learners' (or models) that are particularly well suited to the specific profile of each patient. In this way, the ensemble uses models that are more accurate within particular regions of the feature space, such as models trained on specific age groups or on patients with certain comorbidities. This flexibility to dynamically adjust the ensemble to individual cases often results in better predictive performance than static methods, which apply the same model or fixed subset to all patients.

Finally, in the combination phase, the selected models are combined to generate a final pre-

diction, often through simple methods such as averaging or weighted averaging. In contrast to static weighting, which applies a fixed set of weights based on overall model performance, dynamic weighting adjusts these weights based on the relevance of each model's performance in specific regions of the feature space. This approach has been found to improve accuracy because it allows the ensemble to account for localized patterns in the data. By emphasizing models that perform well within a given region of the feature space, dynamic weighting tailors predictions to the nuances of each test instance, leading to more precise and reliable results (Valdovinos, Sánchez, & Barandela, 2005).

One field where personalization is especially critical is medicine, where predictive models must adapt to the unique characteristics of individual patients. In kidney transplantation, even small improvements in predictive accuracy can affect prognosis and follow-up. The primary dataset in this thesis consists of high-dimensional Nuclear Magnetic Resonance (NMR) spectroscopy data from kidney transplant recipients, which provides rich metabolic profiles but also poses substantial challenges due to its dimensionality and heterogeneity.

To test whether the proposed dynamic ensemble selection (DES) framework generalizes beyond high-dimensional metabolomics, we also evaluate it on a larger, public cohort: the National Health and Nutrition Examination Survey (NHANES) with linked mortality follow-up. NHANES differs from the NMR dataset along two informative axes: it includes substantially more subjects but far fewer features (mostly demographic and clinical variables). This contrast allows us to probe when instance-wise model selection is beneficial (complex, high-dimensional profiles) versus when simpler aggregation may suffice (lower-dimensional, population-scale data).

This thesis investigates DES within ensembles of XGBoost survival models. Unlike static approaches that rely on a fixed subset of models, DES adaptively selects and weights the most competent models for each patient, tailoring predictions to local patterns in the feature space. This patient-specific approach takes advantage of metabolic signatures captured by NMR data and serves as a stress test in NHANES, providing a flexible and clinically meaningful framework for survival prediction.

1.1 Research Question

In light of the preceding discussion, the research questions at the core of this study can be formulated as follows.

- Does dynamic ensemble selection improve the prediction of kidney graft failure from NMR data compared to static selection and uniform averaging methods?
- Which DES strategy—single selection (S), weighting (W), or weighting with selection (WS)—yields the best predictive performance on the NMR dataset?
- Does optimizing DES competence weights with a genetic algorithm improve prediction on NMR?
- **Generalization:** Do the observed DES gains in high-dimensional NMR data transfer to a lower-dimensional, larger cohort (NHANES) or do baseline / static methods perform comparably there?

1.2 Contributions

This thesis makes the following contributions to survival analysis and ensemble learning:

- We implement and evaluate a dynamic model selection framework for survival analysis, applying it to an ensemble of XGBoost models trained on NMR spectroscopy data from kidney transplant recipients. The framework dynamically selects the most competent models per test instance based on local (region-specific) performance.
- We conduct a systematic comparison between dynamic and static ensemble selection strategies, showing when instance-wise selection improves predictive performance over uniform averaging and static pruning.
- We study competence-weighted aggregation and test genetic algorithm (GA) optimization of competence weights, analyzing benefits and failure modes in event-sparse settings.
- **Cross–dataset evaluation:** We replicate the full pipeline on NHANES with linked mortality follow-up to assess generalizability across data regimes (high-dimensional NMR vs. lower-dimensional NHANES), providing guidance on when DES is advantageous versus when simpler ensembles suffice.

1.3 Thesis Outline

- **Chapter 1 – Introduction**: Motivation, problem setting, research questions, and contributions.
- **Chapter 2 – Literature Review**: Core concepts in survival analysis (including censoring), classical models (Cox, AFT), ensemble methods, and model selection (static vs. dynamic) with applications to transplantation and metabolomics.
- **Chapter 3 – Methodology**: Datasets (NMR and NHANES), targets, preprocessing, evaluation metrics, and the proposed DES framework (S, W, WS) including competence measures and GA weighting.
- **Chapter 4 – Experimental Setup**: Train/test splits, cross-validation protocol, baselines, hyperparameters, and a unified cross-data set protocol applied identically to NMR and NHANES.
- **Chapter 5 – Results**: Cross-validation and external test performance on NMR and NHANES, strategy comparisons, GA optimization effects, and additional analyses (e.g., calibration, patient-level timelines).
- **Chapter 6 – Discussion**: Interpretation of findings in data sets, implications for when DES helps, limitations, and methodological considerations.
- **Chapter 7 – Conclusion and Future Work**: Summary of contributions and avenues for scaling, calibration, interpretability, and broader survival settings.

2 Literature Review

2.1 Survival Analysis

Survival analysis is essential in medical studies where the timing of an event, such as graft failure in organ transplantation, is critical. Traditional binary outcome models, such as logistic regression, simply categorize events as occurring or not, which misses crucial information about when the failure occurs. For example, differentiating between graft failure occurring within the first month after transplantation and failure after several years is vital, as it reveals different risk factors and treatment effects.

In survival analysis, the main focus is on the time-to-event data, referred to as the failure time, survival time, or event time. In the event of graft failure, the event would be the failure of the transplanted organ, and the time-to-event would be how long the graft remains functional after transplantation. This method also handles censored data, which occurs when the time of an event of a subject is not fully observed. For example, if a patient in a study is still alive with a functioning graft after five years of follow-up, their time-to-failure is right-censored at five years (denoted as 5+).

A core aspect of survival analysis is the estimation of the survival function, $S(t)$, which gives the probability that the event occurs after time t :

$$S(t) = P(T > t) \tag{1}$$

where T is the time to the event.

2.1.1 Censoring

Censoring occurs because participants are rarely continuously monitored from the moment they become at risk until the event actually occurs. Some enroll late, others miss visits or withdraw, and many remain event-free when the study ends. Consequently, each record provides only a bound on the true event time - after the last contact, before enrollment, or between two assessments - rather than on the exact date. Provided that the censoring mechanism is *non-informative* (that is, not related to future risk given the observed history), these limits can be analyzed with standard survival analysis methods. Figure 1 and Table 1 summarize the four resulting patterns: a fully observed event and the three types of canonical censoring: right, left, and interval, which underpin the modeling strategy used in this thesis.

Table 1: Different types of censoring Harrell (2015)

Type	**Lower bound**	**Upper bound**
Uncensored	T	T
Left-censored	0	Follow-up n
Right-censored	Follow-up n	$+\infty$
Interval-censored	Follow-up $n-1$	Follow-up n

- **Fully observed (Patient 1).** The event occurs during active follow-up, so its exact time is recorded.

- **Right-censoring (Patients 2–3).** The study ends—or the subject is lost –before the event is seen; we know only that the event, if it occurs, occurs *after* the last contact. *Example:* a transplant recipient is still alive at the end of the study.
- **Left-censoring (Patient 4).** The event has begun before the observation starts, leaving only a *upper* bound on its true time. *Example:* a disease detected on a routine exam almost certainly started earlier, but the onset date is unknown.
- **Interval-censoring (Patient 5).** The event is known to fall between two visits, giving a time window rather than a single point. *Example:* a condition screened only yearly must have developed between the last negative and the first positive result.

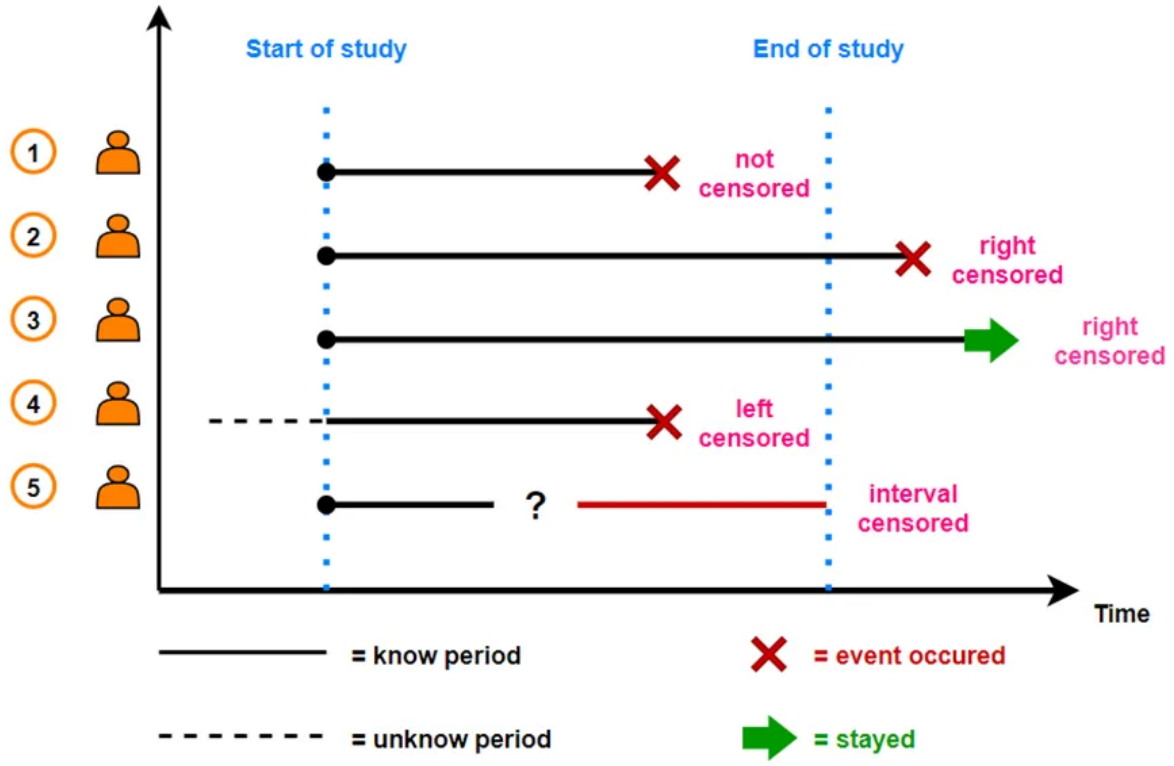

Figure 1: Patient timelines illustrating full observation, right censoring, left censoring, and interval censoring. *Basics of survival analysis in Biostatistics* (n.d.)

2.1.2 Current methodologies

Classical approaches to time-to-event analysis rely on the nonparametric Kaplan-Meier estimator and the semiparametric Cox proportional hazards model (Cox-PH) (Cox, 1972; Kaplan & Meier, 1958). These tools remain workhorses of clinical research but assume linear, additive covariate effects, and (for Cox-PH) proportional hazards. To accommodate nonlinearities and high-dimensional

inputs, researchers increasingly turn to machine learning (ML) methods. A survey of 18 kidney transplant studies by Senanayake et al. (2019) shows that decision trees, Bayesian networks, and neural networks have all been explored, although with mixed improvements over Cox baselines. The remainder of this section therefore reviews:

(i) Cox-PH,

(ii) accelerated-failure-time (AFT) regression, and including associated model-selection challenges.

2.1.3 Semi-parametric Cox proportional-hazards model

The Cox proportional-hazards model (Cox-PH) (Cox, 1972) is one of the most widely used regression approaches in survival analysis. It describes the instantaneous risk of experiencing an event at time t, known as the hazard function $\lambda(t \mid \mathbf{x})$, as a product of a baseline hazard and covariate effects:

$$\lambda(t \mid \mathbf{x}) = \lambda_0(t) \exp(\mathbf{w}^\top \mathbf{x}), \tag{2}$$

In this formulation, $\lambda_0(t)$ represents the baseline risk shared between all individuals, $\mathbf{x}$ denotes the patient's covariate vector that incorporates factors such as age or biomarkers, and $\mathbf{w}$ comprises logarithmic risk ratios that quantify covariate effects on risk.

The model exhibits several defining characteristics. Its semiparametric nature avoids restrictive assumptions about the baseline hazard's temporal pattern while retaining interpretable covariate effects. The exponentiated value of each coefficient ($\exp(w_i)$) yields a hazard ratio that represents the multiplicative effect on risk per unit change in the corresponding covariate. Central to the model's validity is the proportional hazards assumption, mathematically expressed as:

$$\frac{\lambda(t \mid \mathbf{x}_a)}{\lambda(t \mid \mathbf{x}_b)} = \text{constant} \quad \forall t$$

This requirement that hazard ratios remain constant over time represents both the strength and limitation of the model. When violated - such as when treatment efficacy diminishes or biomarker effects evolve during follow-up - risk estimates become biased. Such departures from proportionality can be identified through diagnostic tools including Schoenfeld residuals, which systematically detect time-varying effects, or graphical methods assessing parallelism in log-log survival curves.

Clinically, the model outputs relative rather than absolute risks, necessitating additional steps to predict actual survival times. These relative risk measures, while valuable for comparing patient groups, do not directly translate to clinically actionable time-to-event predictions without additional baseline hazard estimation. When proportionality violations emerge, methodological adaptations, such as stratification or time-varying coefficients, become essential to maintain model validity and clinical interpretability.

2.1.4 Parametric Accelerated Failure Time (AFT)

The Accelerated Failure Time (AFT) model is another established approach to survival analysis, offering an alternative to Cox-PH. Unlike Cox, which models relative risk, AFT directly models survival time by assuming that covariates accelerate or decelerate the life course by a constant factor. The model is written as:

$$\ln Y = \langle \mathbf{w}, \mathbf{x} \rangle + \sigma Z, \tag{3}$$

where Y is the survival time, $\mathbf{x}$ the covariates, $\mathbf{w}$ the coefficients, and Z a random variable drawn from a specified distribution. Common distributional choices for Z include exponential, Weibull, or lognormal, each leading to different parametric forms of the survival function.

A key advantage of AFT is its ability to provide direct estimates of survival times and absolute risks, making it more interpretable in certain clinical settings. Moreover, it does not rely on the proportional hazards assumption, which makes it attractive when relative risks vary over time.

However, the model is limited by its reliance on a correct specification of the survival time distribution. If the chosen distribution does not accurately reflect the underlying data, the estimates may be biased or unstable. This sensitivity to distributional assumptions explains why AFT is less commonly applied in practice than Cox-PH, despite its theoretical appeal. Extensions using penalization (for example, elastic net regularization) and integration into boosting frameworks have improved its applicability to high-dimensional data Khan and Shaw (2016).

2.2 Machine-learning and ensemble methods

Survival trees and forests. A survival tree partitions the covariate space using logarithmic rank or likelihood splits; a *Random Survival Forest* (RSF) aggregates many such trees grown on bootstrap samples, averaging their cumulative-hazard estimates Breiman (2001); Pickett et al. (2021). RSF handles non-linear interactions and missing data with minimal tuning.

Boosted survival models. Gradient boosting builds an additive ensemble of weak learners (e.g., decision trees), each sequentially optimized to minimize survival-specific loss functions. These include the Cox partial likelihood for hazard ratios or smoothed concordance for classification accuracy, yielding strong discrimination while retaining variable importance measures Y. Chen, Jia, Mercola, and Xie (2013); Khan and Shaw (2016); Pölsterl (2020).

Stacking and heterogeneous ensembles. While bagging reduces variance and boosting reduces bias, large homogeneous ensembles may overfit. Stacked generalization (e.g., super-learners) combines heterogeneous base models, such as Cox-PH, random survival forests, and gradient-boosted learners via a meta-learner trained on out-of-fold predictions. This integrates complementary strengths and often outperforms any single constituent Hu, Ji, and Li (2021); Yan and Feng (2022).

Although ensemble methods consistently improve predictive performance, their use in survival analysis also introduces practical challenges. Large ensembles often contain many redundant or weak learners, and their predictive value can vary substantially between different patient subgroups. This raises the question *model selection*: Which subset of learners should be retained and how should they be combined to achieve robust, efficient, and interpretable predictions in survival settings? We therefore turn to model selection as a central focus of this thesis.

2.3 Model Selection in Survival Ensembles

As ensemble learning methods have gained prominence in survival analysis, the problem of *model selection* has emerged as a central challenge. Ensembles are typically constructed by aggregating a large number of base learners, such as decision trees, Cox models, or gradient-boosted regressors, trained on resampled or partitioned subsets of the data. Although this strategy increases predictive accuracy and robustness, it also introduces redundancy: many base models may contribute little to overall performance, while others may perform well only for specific patient subgroups. Therefore, selecting the most relevant models, globally or locally, is crucial to balancing predictive performance, computational efficiency, and clinical interpretability (**??**).

Static model selection. In static approaches, a subset of models is chosen prior to deployment and subsequently applied to all future patients. Selection can be achieved through partitioning-based methods, such as clustering similar learners and retaining representatives from each cluster (Coelho & Von Zuben, 2006; Lazarevic & Obradovic, 2001), or through search-based strategies, including greedy selection (Partalas, Tsoumakas, Hatzikos, & Vlahavas, 2008), genetic algorithms, and tabu search (Ruta & Gabrys, 2001). Although effective in reducing redundancy, static methods disregard instance-specific differences: a model that performs well for one patient may be suboptimal for another.

Dynamic model selection. Dynamic methods address this limitation by tailoring the selection of models to each test instance. The concept originates from the definition of a *region of competence* around the query point, typically identified using nearest neighbors in the training set (Woods, Kegelmeyer, & Bowyer, 1997). Within this local neighborhood, base models are ranked or weighted according to their competence, estimated from historical errors or other performance measures. Several dynamic strategies have been proposed:

- **DS** — dynamic selection of the single most competent model for the query instance (Rooney, Patterson, & Nugent, 2004).
- **DW** — dynamic weighting of all models, where weights reflect local competence (Mendes-Moreira, Jorge, Soares, & de Sousa, 2009).
- **DWS** — a hybrid method that prunes underperforming models and then applies competence-based weighting to the remainder (Mendes-Moreira, Soares, Jorge, & Sousa, 2015).

Dynamic selection has been shown to improve predictive accuracy in heterogeneous domains, but it also raises challenges regarding computational cost, stability, and the definition of reliable competence measures in censored survival data.

Open challenges. Despite advances in both static and dynamic selection, three persistent issues remain. First, there is no consensus on how to define and measure competence in survival analysis, particularly under censoring. Second, most methods have been developed for classification and regression tasks, with relatively little attention paid to time-to-event outcomes. Finally, external validation in medical applications remains rare, limiting the generalizability of findings. Addressing

these gaps requires frameworks that not only adapt model selection to the patient level, but also integrate survival-specific performance metrics and evaluate generalization on independent cohorts.

In this thesis, we build upon these insights by developing and benchmarking dynamic ensemble selection strategies tailored to survival prediction. By applying these methods to kidney transplant outcomes and high-dimensional NMR metabolomics data, we aim to demonstrate the potential of competence-aware model selection to enhance predictive accuracy in clinically heterogeneous settings.

2.4 Applications in Medicine

2.4.1 Survival Prediction of Graft kidney failure

Kidney transplantation is the preferred treatment for patients with end-stage renal disease, offering better survival and quality of life compared to dialysis. However, graft failure remains a major concern, as it necessitates a return to dialysis or re-transplantation and is associated with increased morbidity and mortality. The causes of graft failure are complex and multifactorial, including immunological rejection, recurrence of primary kidney disease, cardiovascular complications, and infections. Despite advances in immunosuppressive therapy, long-term graft survival has only modestly improved over the last decades, making the accurate prediction of graft failure a critical goal in transplantation medicine Meier-Kriesche, Schold, and Kaplan (2004); Meier-Kriesche, Schold, Srinivas, and Kaplan (2004).

Traditionally, prognostic models for graft survival have relied on clinical and demographic risk factors such as donor and recipient age, HLA mismatching, cold ischemia time, and delayed graft function. Statistical methods such as logistic regression and the Cox proportional hazards model have been applied extensively, but they are limited by their assumptions of linearity and proportional hazards, which often fail to capture the heterogeneity of transplant populations Senanayake et al. (2019). As a result, the predictive accuracy of conventional models remains modest, particularly for long-term outcomes where multiple interacting factors influence graft survival.

In recent years, machine learning approaches have been explored as alternatives for predicting graft failure. A systematic review by Senanayake et al. (2019) evaluated 18 studies applying machine learning to kidney transplant outcomes. Cohort sizes varied widely, from fewer than 100 patients to nationwide registries with more than 90,000 recipients, reflecting the heterogeneity of available datasets. Popular methods included decision trees, artificial neural networks, and Bayesian belief networks. Although these models sometimes outperformed traditional regression in terms of discrimination, the results were inconsistent and, in some cases, simpler models performed equally well. In particular, only a minority of studies incorporated survival modeling, despite its clear relevance to event-time outcomes. Most machine learning studies instead focused on classification tasks (e.g., graft survival at 1 year), which disregards valuable information about the timing of failure.

Another issue highlighted by Senanayake et al. (2019) was methodological rigor. Few studies performed external validation, many did not report hyperparameters, and there was substantial variability in the choice of predictors, ranging from basic demographic data to complex immunological and laboratory variables. This heterogeneity makes comparison between studies difficult and underscores the need for standardized methodologies.

In general, prediction of survival for kidney graft failure remains an open challenge. Although machine learning methods offer promise, their success has so far been limited by small sample

sizes, inconsistent methodologies, and insufficient use of survival data. There is a clear need for models that can better capture nonlinear interactions, adapt to heterogeneous patient populations, and integrate high-dimensional biomarker data. Addressing these challenges would not only improve the accuracy of graft failure prediction but would also support more personalized approaches to post-transplant care.

2.5 Nuclear Magnetic Resonance (NMR)

Nuclear Magnetic Resonance (NMR) spectroscopy has become an important tool for machine learning–based medical prediction Corsaro et al. (2022). From a plasma sample, the instrument records radiofrequency signals as the nuclei return to equilibrium in a magnetic field. These raw signals, called free induction decays (FIDs), are Fourier transformed into frequency-domain spectra. Each spectrum contains distinct peaks reflecting the chemical environment of atoms, enabling the molecular characterization of biological fluids such as blood or plasma.

This application, known as metabolomics, provides valuable information on biomarkers and disease mechanisms Johnson, Ivanisevic, and Siuzdak (2016); Nicholson, Buckingham, and Sadler (1983). Machine learning has been widely used in this context, from signal processing to biomarker discovery, with promising results for conditions such as cancer relapse and Alzheimer's disease Cobas (2020); Di Donato et al. (2021). For example, Peng, Ng, and Loh (2020) showed that two-dimensional NMR data could be used for rapid blood phenotyping and accurate diagnosis from just a drop of blood. NMR-based assays are thus cost-effective, minimally invasive, and fast.

However, a major challenge is the very high dimensionality of NMR spectra, which complicates both computational analysis and interpretability. Dimensionality reduction is often used Costanti, Kola, Scarselli, Valensin, and Bianchini (2023), but this risks discarding useful information. Survival analysis using NMR data remains scarce for this reason. Deelen et al. (2019), for instance, applied a Cox-PH model to hand-selected metabolic biomarkers to predict all-cause mortality, but this targeted approach risks overlooking novel predictors. Similarly, Kaynar et al. (2023) reduced spectra to metabolites using a neural network before survival modeling, improving accuracy but limiting discovery to predefined metabolites.

These challenges make NMR-based survival prediction an ideal testing ground for advanced model selection strategies. Selecting and weighting the most informative learners dynamically allows the full richness of spectral data to be leveraged without discarding potentially valuable signals, offering a path toward more accurate and interpretable survival models in metabolomics.

3 Methodology

3.1 Dataset

This study evaluates the proposed framework on two cohorts with complementary characteristics: (i) a high–dimensional metabolomics dataset from the ELSE laboratory (plasma NMR spectroscopy with clinical follow-up in renal transplant recipients), and (ii) a larger, lower-dimensional public cohort from the National Health and Nutrition Examination Survey (NHANES) with linked mortality follow-up. Both datasets are analyzed under a time-to-event formulation with right censoring.

3.1.1 ELSE Lab Clinical Cohort

Clinical data is derived from a prospective longitudinal cohort in the Northern Netherlands, collected within the TransplantLines Food and Nutrition Biobank and Cohort study (NCT02811835) between November 2008 and June 2011 van den Berg et al. (2014). All renal transplant recipients aged $\geq$18 years with at least one year of functional allograft were eligible. Exclusion criteria included congestive heart failure, cancers other than cured skin cancer, and endocrine disorders other than diabetes mellitus. Of 817 eligible patients, 707 (86.5%) provided informed consent, without material differences from the broader eligible population. Patients with missing values required to define remnant cholesterol or new-onset diabetes after transplantation (NODAT) (n=57), or with a history of diabetes or glucose-lowering medication (n = 169), were excluded, producing 480 RTR eligible for analysis in the source study. The study was approved by the local Institutional Review Board (METc 2008/186) and conducted in accordance with the Declaration of Helsinki.

For the present work, we analyzed the subset of RTR with available plasma NMR spectra and complete survival follow-up throughout the fourth scheduled visit. After matching clinical records with spectroscopy availability, the resulting analytical cohort comprised 249 patients. Patients without an event at the fourth follow-up were right-censored at that visit.

3.1.2 NMR Spectroscopy Data

Plasma NMR spectra were acquired on a Varian spectrometer operating at a proton resonance frequency of 400 MHz, with analyses performed on a Vantera® Clinical Analyzer. Each spectrum contained 32,768 points over a spectral width of 4,496 Hz. A Carr–Purcell–Meiboom–Gill (CPMG) pulse sequence was used to emphasize low molecular weight metabolites, and free induction decay (FID) signals were converted to frequency domain spectra via Fourier transform.

Preprocessing retained only the absorption (real) component (spectra were properly phased), applied baseline correction by subtracting average noise estimated from the first and last 1,000 points, and averaged the three replicate spectra per sample to improve signal-to-noise ratio. Noisy spectral regions at the extremes (chemical shifts $>$9.0 ppm and <-0.5 ppm) were removed. The final preprocessed NMR matrix comprised 18,724 spectral intensity features per patient for the 249 matched RTR.

3.1.3 NHANES Mortality Cohort

NHANES is a continuous, nationally representative health survey of the U.S. population, fielded in two-year cycles since 1999 (Centers for Disease Control and Prevention (n.d.)). It combines in-home

interviews with standardized examinations in mobile examination centers (MECs). For this study, we used publicly available NHANES data with the National Center for Health Statistics (NCHS) Linked Mortality File. The *target* was time-to-all-cause mortality; participants without a recorded death were right-censored at their last known follow-up in the linkage file. Cases without valid follow-up time were excluded.

NHANES includes substantially more participants than the NMR cohort but far fewer predictors, primarily demographic, examination, and laboratory variables. This contrast provides a complementary testbed for evaluating the proposed method in a lower-dimensional, population-scale setting.

NHANES preprocessing. Predictors were kept in their native numeric form; boolean variables were cast to $\{0, 1\}$. Missing predictor values were imputed using the column median. No additional scaling, normalization, or dimensionality reduction was applied.

3.1.4 Descriptive Analysis (NMR)

A descriptive analysis of the final dataset was conducted. The dataset comprises 249 samples, of which 30 (12.05%) experienced graft failure or patient death (see Figure 3). The average follow-up time for all participants was approximately 6.47 years (SD = 2.28), with event times ranging from 0.20 to 9.08 years. The median survival time was 7.38 years, indicating a high degree of right-censorship.

To assess and visualize the temporal dynamics of graft survival, we constructed a Kaplan–Meier (KM) survival curve using observed time-to-event data with censoring properly accounted for. The KM estimate in Figure 2 shows the probability of graft survival over time. The curve starts at 1.0 (indicating 100% survival) and gradually decreases at the times of graft failure events. The shaded region around the survival function denotes the 95% confidence interval. A more pronounced drop is visible around 9 years, corresponding to a cluster of events, suggesting limited long-term durability for some grafts. The relatively high proportion of censored observations, particularly in later follow-up, contributes to the widening of the confidence interval over time.

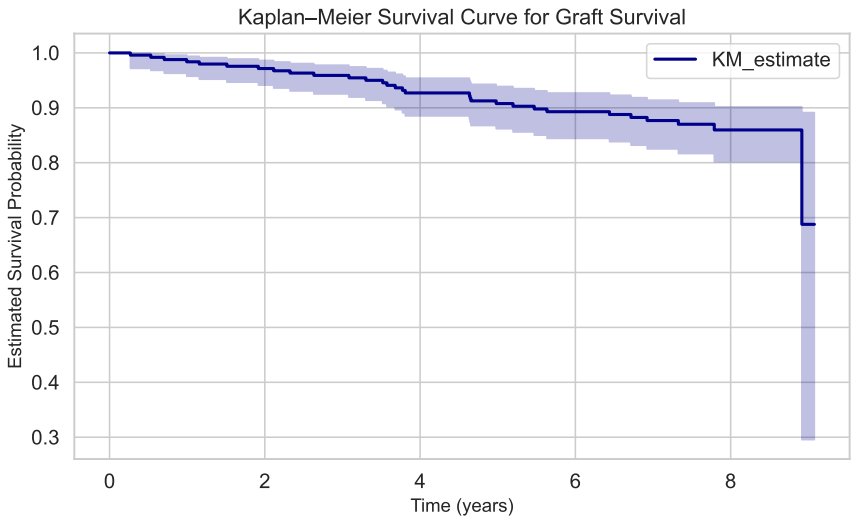

Figure 2: Kaplan–Meier survival curve for kidney graft survival in the NMR dataset. The curve represents the estimated probability of survival over time, with the shaded area indicating the 95% confidence interval. The gradual decline reflects steady event occurrences, while the sharp drop near 9 years indicates a cluster of late graft failures. Right censoring contributes to increased uncertainty in later time points.

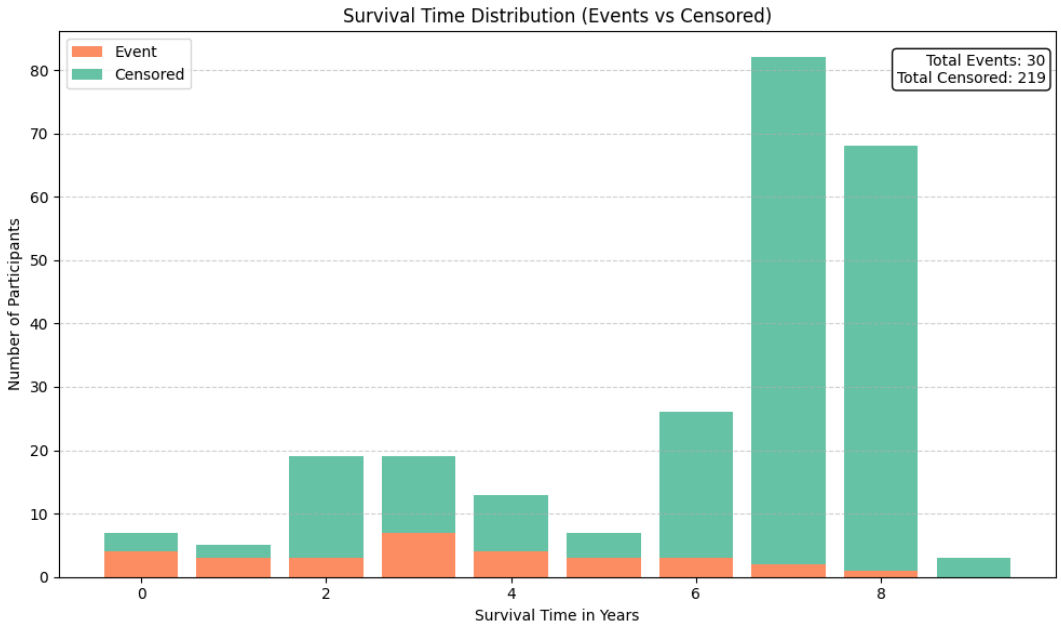

Figure 3: Survival time distribution of participants in the NMR dataset. Bars distinguish between participants who experienced the event (graft failure or death) and those who were censored. A total of 30 events were observed, while 219 participants were censored.

3.1.5 Data Split

To ensure fair model evaluation and prevent data leakage, the dataset was partitioned into a training/validation set and a holdout test set. As shown in Figure 4, the original dataset of 249 samples was split into:

- **Training/Validation Set (80%)**: 199 samples, including 25 observed events. This subset was used for model training, dynamic ensemble selection, weight optimization, and cross-validation.
- **Test Set (20%)**: 50 samples, including 5 observed events. This subset was kept aside and only used once for the final evaluation of the models trained and optimized in the training / validation set.

Within the training/validation subset, cross-validation was performed to evaluate model selection strategies and to optimize competence weights in dynamic ensemble selection using genetic algorithms. The final trained model (or ensemble) was then applied to the held-out test set to assess generalization performance. This unified protocol was applied identically to NMR and NHANES.

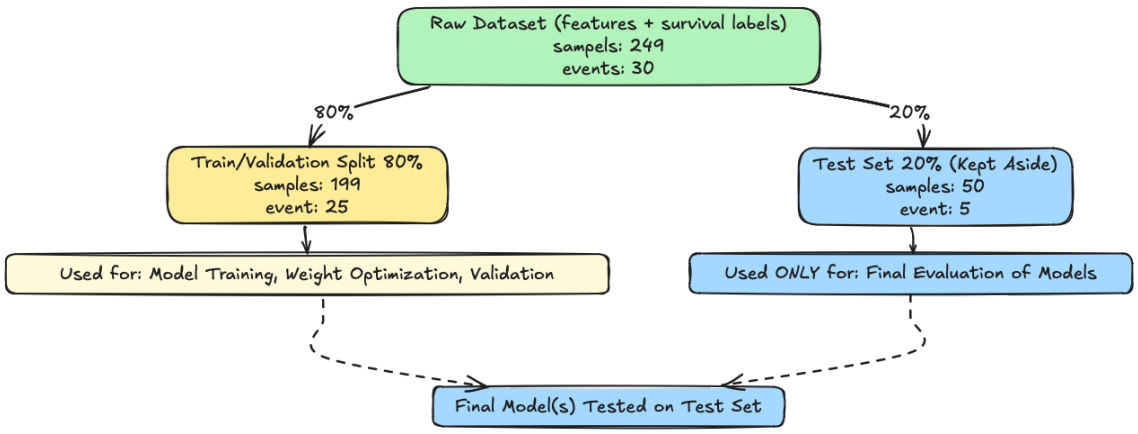

Figure 4: Data split schema for model training, validation, and final evaluation.

3.2 Metrics

3.2.1 C-index

The primary evaluation metric used to assess model performance in this survival analysis study is the concordance index (C-index). The C-index quantifies the predictive capability of a model by measuring its ability to correctly order pairs of individuals based on predicted survival times relative to their actual observed survival times.

Specifically, the C-index evaluates pairs of samples, determining whether the predicted survival times correctly reflect which individual survives longer. Concordant pairs occur when the model correctly predicts the relative order of survival durations between two individuals, while discordant pairs arise when the predicted ordering contradicts the observed outcomes. If the model predicts

identical survival times for a pair, the pair is considered tied and is given half the weight of a concordant pair.

The C-index is calculated using the following formula:

$$C = \frac{\text{Number of Concordant Pairs} + 0.5 \times \text{Number of Tied Pairs}}{\text{Total Number of Comparable Pairs}} \tag{4}$$

It is important to note that the calculation of the C-index only considers pairs involving at least one uncensored observation. Pairs consisting solely of censored observations are excluded, as they provide no definitive outcome comparison. Additionally, comparisons involving censored observations can only be made if the observed event time for the uncensored observation occurs before the censoring time of the censored observation.

Thus, the C-index provides an effective measure of the predictive discrimination of survival models, indicating the extent to which the models can correctly rank individuals by their risk of experiencing the event of interest.

3.3 Framework

This study employs a dynamic selection framework specifically designed for survival analysis, utilizing multiple error-based competence measures. As illustrated in Figure 5 the architecture comprises three sequential phases: Generation, Optimization, and Generalization.

- **Generation**: Constructs a homogeneous ensemble of XGBoost models using the Accelerated Failure Time (AFT) approach, followed by computation of survival-specific competence measures
- **Optimization**: Assigns weights to competence measures reflecting their predictive importance
- **Generalization**: Dynamically selects ensemble subsets using weighted competence scores to generate predictions for query samples

3.3.1 Ensemble generation

In this phase, we generate a homogeneous ensemble $\mathcal{F}$ comprising N XGBoost models, each configured with the Accelerated Failure Time (AFT) loss function for survival analysis. Using the Bagging algorithm (Bootstrap Aggregating) Breiman (2001), each model is trained on distinct bootstrapped subsets of the dataset. This approach induces model diversity and enhances predictive robustness. The resulting ensemble members subsequently serve as candidate models for dynamic selection during prediction.

3.3.2 Optimization

Dynamic regressor selection identifies query-specific optimal models by evaluating prediction errors in local competence regions. Six errors were assessed, which capture distinct performance dimensions under survival constraints. In the absence of a universally optimal measure, we integrate them using a weight vector $\mathcal{W} = w_1, \ldots, w_6$. This phase involves: (1) extracting competence measures and (2) optimizing their weights.

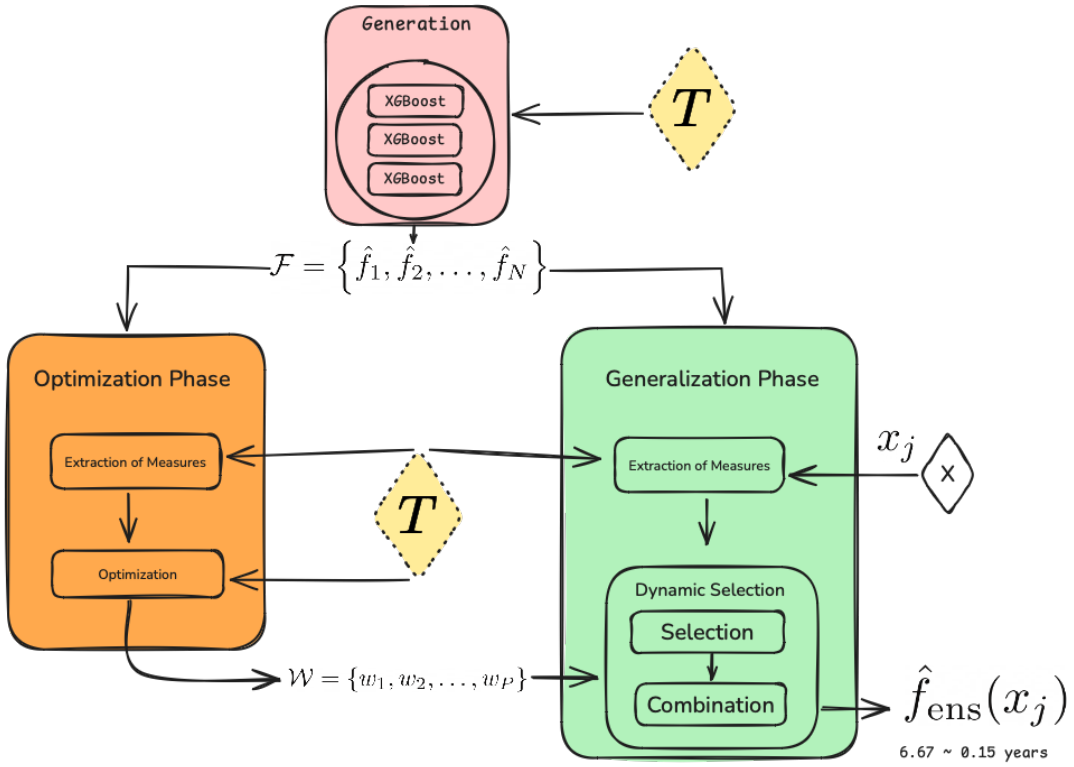

Figure 5: Overview of the Dynamic Selection Framework. The pipeline consists of three main phases: generation, optimization, and generalization. A homogeneous ensemble $\mathcal{F}' = \{\hat{f}_1, \ldots, \hat{f}_N\}$ is trained on bootstrapped samples from the training set $\mathcal{T}$. Competence measures are extracted and optimized to obtain a weight vector $\mathcal{W} = \{w_1, \ldots, w_P\}$. During generalization, for each test instance $x_j \in \mathcal{X}$, the most competent models are dynamically selected and combined to produce the final ensemble prediction $\hat{f}_{\text{ens}}(x_j)$.

Legend: $\mathcal{T}$ – Training set; $\mathcal{X}$ – Test set; $\mathcal{F}'$ – Full ensemble of base learners; $\mathcal{F}$ – Selected subset of models; $\mathcal{W}$ – Optimized competence weights; x_j – Test instance; $\hat{f}_{\text{ens}}(x_j)$ – Final ensemble prediction.

Extraction of Error Measures A total of six error-based competence measures $\{m_1, m_2, \ldots, m_6\}$ are extracted from the region of competence for each regressor $\hat{f}_n \in \mathcal{F}$. These measures are designed to capture different aspects of the quality of prediction in survival analysis.

To compute neighborhood-based measures, the distance weight d_k for a neighbor pattern t_k is defined as the inverse of its normalized distance:

$$d_k = \frac{1/\text{dist}_k}{\sum_{j=1}^{K} 1/\text{dist}_j} \tag{5}$$

where $\{\text{dist}_1, \text{dist}_2, \ldots, \text{dist}_K\}$ are the distances between the test pattern and its K nearest neighbors in the training set $\mathcal{T}'$. Each distance dist_k corresponds to the similarity between a query instance and a neighbor pattern t_k, with $k \in \{1, 2, \ldots, K\}$. Normalizing the distances in this way ensures that the weights d_k lie in the interval $[0, 1]$ and sum to 1, giving more influence to closer neighbors during competence estimation. The region of competence $\Psi = \{t_1, t_2, \ldots, t_K\}$ is defined for each test sample and used to calculate the error-based competence of each regressor $\hat{f}_n$. The six competence measures

Table 2: Error–based competence measures used for dynamic ensemble selection in survival analysis.

Measure	**Acronym**	**Equation**
Variance	m_1	$\mathrm{Var}\big(\hat{f}_n(t_1), \hat{f}_n(t_2), \ldots, \hat{f}_n(t_K)\big)$
Prediction Range	m_2	$\max\limits_{1\le k\le K} \hat{f}_n(t_k) - \min\limits_{1\le k\le K} \hat{f}_n(t_k)$
Prediction Consistency	m_3	$\frac{1}{K}\sum\limits_{k=1}^{K} \left\lvert \hat{f}_n(t_k) - \hat{f}_n(x_j) \right\rvert$
Censored–Weighted Squared Error	m_4	$\frac{\sum_{k=1}^{K} e_k \left(\hat{f}_n(t_k) - y_k\right)^2}{\sum_{k=1}^{K} e_k}$
AFT Negative Log-Likelihood Proxy	m_5	$\frac{1}{K}\sum\limits_{k=1}^{K} \left[\frac{1}{2}\left(\frac{y_k - \hat{f}_n(t_k)}{\sigma} \right)^2 + \frac{1}{2}\log\big(2\pi\sigma^2\big) \right]$
Discordance Error	m_6	$\sum\limits_{k=1}^{K} d_k \, n_{k,n}^{\text{discordant}}$

used in this framework are described below.

- m_1 - **Variance:** Measures the prediction spread within the region of competence. The variance is calculated for each regressor using the estimated values for the patterns in the region of competence, according to Eqquation 6:

$$m_{1,n} = \mathrm{Var}(\hat{f}_n(t_1), \hat{f}_n(t_2), \ldots, \hat{f}_n(t_K)) \tag{6}$$

- m_2 - **Prediction Range:** Measures the spread of predictions made by regressor $\hat{f}_n$ within the region of competence. It captures the difference between the maximum and minimum predicted values among the K nearest neighbors of a test instance. A larger range may indicate higher uncertainty or model instability in that local region.

  The prediction range $m_{2,n}$ is computed as:

$$m_{2,n} = \max\left(\hat{f}_n(t_1), \hat{f}_n(t_2), \ldots, \hat{f}_n(t_K)\right) - \min\left(\hat{f}_n(t_1), \hat{f}_n(t_2), \ldots, \hat{f}_n(t_K)\right) \tag{7}$$

  Here, $t_1, t_2, \ldots, t_K$ represents the K patterns in the region of competence, and $\hat{f}_n(t_k)$ denotes the prediction made by regressor $\hat{f}_n$ on sample t_k.

- m_3 - **Prediction Consistency:** This measure quantifies how consistent the predictions of regressor $\hat{f}_n$ are across the region of competence $t_1, t_2, \ldots, t_K$ compared to its prediction on the test instance x_j. A lower value indicates that the model behaves similarly in the neighborhood and on the test sample, suggesting more reliable local behavior.

  The prediction consistency $m_{3,n}$ is defined as:

$$m_{3,n} = \frac{1}{K} \sum_{k=1}^{K} \left| \hat{f}_n(t_k) - \hat{f}_n(x_j) \right| \tag{8}$$

Here, $\hat{f}_n(t_k)$ is the prediction of regressor $\hat{f}_n$ on the k-th neighbor t_k, $\hat{f}_n(x_j)$ is the prediction of the same regressor on the test pattern x_j, and K is the number of nearest neighbors in the region of competence.

This metric penalizes models that produce locally unstable or erratic predictions relative to the test sample.

- m_4 - **Censored-Weighted Squared Error:** This measure emphasizes prediction accuracy on uncensored samples, which provide complete event-time information. It applies higher penalties to errors where the true event time is known (i.e., uncensored), while assigning less weight to censored observations.

  The censored-weighted squared error $m_{4,n}$ is computed as:

$$m_{4,n} = \frac{\sum_{k=1}^{K} e_k \left(\hat{f}n(t_k) - y_k \right)^2}{\sum k = 1^K e_k} \tag{9}$$

  Where y_k is the observed survival time (lower bound) of neighbor t_k, $e_k \in 0, 1$ is the event indicator (1 if uncensored, 0 if censored), $\hat{f}_n(t_k)$ is the prediction of regressor $\hat{f}_n$ on t_k, K is the number of neighbors in the region of competence.

  By weighting the squared error by the event indicator, this measure ensures that uncensored samples—those contributing the most information—dominate the loss computation.

- m_5 - **AFT Negative Log-Likelihood Proxy:** This measure approximates the log-likelihood of the Accelerated Failure Time (AFT) model by assuming normally distributed residuals. It captures how well the model's predictions align with the observed survival times, penalizing large deviations.

  The AFT negative log-likelihood proxy $m_{5,n}$ is defined as:

$$m_{5,n} = \frac{1}{K} \sum_{k=1}^{K} \left[\frac{1}{2} \left(\frac{y_k - \hat{f}_n(t_k)}{\sigma} \right)^2 + \frac{1}{2} \log(2\pi\sigma^2) \right] \tag{10}$$

  Where y_k is the observed survival time of neighbor t_k, $\hat{f}_n(t_k)$ is the prediction of regressor $\hat{f}_n$ for t_k, σ is a fixed scale parameter (commonly set to 1 in this approximation), and K is the number of nearest neighbors in the region of competence.

  This proxy penalizes predictions that deviate substantially from the observed times, consistent with the log-likelihood of the normal distribution used in AFT models.

- m_6 - **Discordance Error:** The discordance error quantifies how frequently a given sample appears in discordant prediction pairs, a concept derived from the concordance index used in survival analysis. A pair of samples is defined as discordant when the model predicts a higher

survival time (i.e., lower risk) for a sample that actually experienced the event earlier—thus misaligning with the observed outcome. This error emphasizes incorrect risk ordering, especially in censored datasets where individual prediction loss is not well-defined.

Let $n_{i,n}^{\text{discordant}}$ denote the number of discordant pairs that sample i is involved in for regressor $\hat{f}n$. The discordance error $m6,n$ for a test instance x_j is then calculated by aggregating these discordance counts across the K nearest neighbors, weighted by their normalized distance:

$$m_{6,n} = \sum_{k=1}^{K} d_k \cdot n_{k,n}^{\text{discordant}} \tag{11}$$

Here, d_k is the normalized inverse distance weight of neighbor t_k, as stated before. This weighting scheme ensures that discordance counts from closer neighbors contribute more strongly to the overall error, thereby focusing the competence evaluation on the local region around the test sample. High values of $m_{6,n}$ indicate that the model frequently ranks survival times incorrectly for nearby patterns and may need to be downweighted during dynamic selection.

For every pair consisting of a training sample $t_i \in \mathcal{T}$ and a regressor $\hat{f}n$, six competence measures are computed based on the region of competence around t_i. These values form a vector $Mi,n = \{m_{1,n}, m_{2,n}, \ldots, m_{6,n}\}$, where each $m_{p,n}$ represents the score of the p-th measure for the regressor $\hat{f}_n$.

GA Optimization The measures extracted in the previous step provide varying degrees of insight into model performance. Therefore, this phase aims to determine optimal weights for each measure, reflecting their relative importance. A Genetic Algorithm (GA) Eiben and Smith (2015) is employed to compute one weight per measure using the vectors $M_{i,n}$ described in the previous section.

Algorithm 1 presents the pseudocode for the optimization procedure. It outputs the weight vector $W = \{w_1, w_2, ..., w_p\}$ that minimizes the C-Index of the training set T. The mutation, crossover, and elitism parameters are detailed in Section 4.4

Algorithm 1: Genetic Algorithm for Dynamic Ensemble Selection (DES)

Input: Trained Ensemble $\mathcal{F}$, Data $\mathcal{X}, \mathcal{Y}$, Parameters: L (population size), G (max generations), K (top elites), τ (c-index threshold)
Output: $\mathbf{w}^*$: Best weight vector

```
1  Initialize population P of L individuals (random, all-ones, one-hot);
2  best_cindex ← −∞;
3  stall ← 0;
4  for g ← 1 to G do
5      Select subset (X_g, Y_g) for this generation;
6      Compute mutation rate μ_g (decaying);
7      foreach individual w in P (in parallel) do
8          Set DESRegression weights w;
9          foreach sample x_i in X_g do
10             Compute competence scores (all models, using kNN);
11             Dynamic selection: select models, aggregate prediction;
12         end
13         Compute c-index of predictions for w;
14     end
15     Save best K elites from P by c-index;
16     Generate new offspring from elites (crossover, mutation) to form new P;
17     Update best_cindex, best_weights if improved;
18     if best_cindex ≥ τ or stall ≥ limit then
19         break;
20     end
21 end
22 return best_weights;
```

The algorithm starts by initializing a population P of L individuals, each representing a candidate weight vector (Line 1). This initialization includes random vectors, an all-ones vector (equal weighting), and one-hot vectors to encourage early diversity. The current best c-index score and a stall counter are initialized (Lines 2–3).

The algorithm then proceeds iteratively for a maximum of G generations (Line 4). In each generation, a bootstrap subset of the training data $(\mathcal{X}_g, \mathcal{Y}_g)$ is sampled (Line 5) to compute fitness scores efficiently. A decaying mutation rate μ_g is computed based on the generation number (Line 6), allowing for larger exploratory mutations in early generations and smaller, fine-tuning mutations in later ones.

Each individual in the population is evaluated in parallel (Line 7). For every weight vector $w \in P$, the 'DESRegression' model is updated to use w as its competence measure weights. Then, for each sample x_i in the training set, the algorithm calculates the competence scores using the selected measures and applies dynamic selection to choose a subset of models from the ensemble (lines 8–10). The predictions of these models are aggregated, and the c-index of the resulting predictions is computed as the fitness of w (Line 11).

After all individuals are evaluated, the top K elites with the highest c-index scores are retained (Line 15). These elites are then used to generate the next population through crossover and mutation (Line 16). Crossover blends weights from two parents, while mutation introduces random

perturbations to promote exploration.

The algorithm tracks the best-performing weight vector and its c-index score across generations. If the current generation yields a better score, it updates the global best; otherwise, it increments the stall counter. The search terminates early if the best c-index exceeds the threshold τ or the stall counter reaches a predefined limit (Line 18).

Finally, the algorithm returns the best-performing weight vector $\mathbf{w}^*$, which is then used in the generalization phase.

3.3.3 Generalization

In the Generalization Phase, the final prediction $\hat{f}_{\text{ens}}(x_j)$ is computed for each test sample $x_j \in \mathcal{X}$ by dynamically selecting and combining regressors from the ensemble $\mathcal{F}$. This phase involves two main steps: *Measure Extraction* and *Dynamic Selection*.

The *Measure Extraction* step defines the region of competence for each test instance x_j using the validation set $\mathcal{T}'$, and computes a vector of local performance measures for each regressor $\hat{f}_n \in \mathcal{F}$, as detailed in Section **??**. The result is a set of measure vectors $\mathbf{M}_{j,n} = \{m_{1,n}, m_{2,n}, \ldots, m_{P,n}\}$ for every test-regressor pair.

The *Dynamic Selection* step evaluates the competence of each regressor $\hat{f}_n$ on x_j by computing a competence score α_n using the optimized weights $\mathbf{w}^*$ obtained during the Optimization Phase:

$$\alpha_n = \sum_{p=1}^{P} w_p \cdot m_{p,n} \tag{12}$$

These competence scores form a vector $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_N)$ used to guide the selection and combination of regressors.

Selection and Combination Strategies Based on the competence scores, three dynamic strategies are implemented in the framework:

- **S**: Selects the single most competent regressor $\hat{f}_n$ with the lowest α_n.
- **W**: Combines all regressors in the ensemble, weighting their outputs inversely to α_n.
- **WS**: Selects a subset of top-k regressors based on α_n and combines their predictions.

The final prediction $\hat{f}_{\text{ens}}(x_j)$ is computed by aggregating the selected regressors' outputs, either through simple averaging or weighted combination depending on the strategy.

These strategies build upon concepts from Mendes-Moreira et al. (2009) but are adapted for survival analysis with a different competence definition based on error measures, and a genetic algorithm-optimized weighting scheme. The framework remains flexible and can be extended to support alternative selection mechanisms.

S Competence measures $m_{p,n}$ quantify distinct aspects of prediction error for regressor $\hat{f}_n$ within its region of competence. A lower aggregated score α_n (weighted sum of these measures) indicates higher regressor competence. The S strategy therefore selects the regressor $\hat{f}_n$ that minimizes α_n for each test instance $x_j \in \mathcal{X}$, as formalized in Equation 13:

$$\text{index} = \arg \min_{1 \leq n \leq N} \alpha_n \tag{13}$$

The final prediction for x_j is then computed using the selected regressor:

$$\hat{f}_{\text{ens}}(x_j) = \hat{f}_{\text{index}}(x_j) \tag{14}$$

where $\hat{f}_{\text{index}}(x_j)$ denotes the prediction from the optimal regressor. The complete procedure is detailed in Algorithm 2.

Algorithm 2: Selecting using S (Single best model)

Input: Ensemble $\mathcal{F}$, Training set $\mathcal{T}'$, Test set $\mathcal{X}$, Weights $\mathcal{W}$, Neighborhood size K
Output: c-index: Concordance Index

Initialize empty lists Y_{true} and Y_{pred};
for *each test pattern* x_j *in* $\mathcal{X}$ **do**
  Find the region of competence Ψ of x_j using $\mathcal{T}'$;
  for *each* f_n *in* $\mathcal{F}$ **do**
    Calculate the measures $\{m_{1,n}, m_{2,n}, ..., m_{P,n}\}$ using Ψ;
    $\alpha_n = \sum_{p=1}^{P} w_p \times m_{p,n}$;
  end
  $index = \arg\min_{1 \leq n \leq N}(\{\alpha_1, ..., \alpha_N\})$;
  $\hat{f}_{\text{ens}}(x_j) = f_{index}(x_j)$;
  Append true value $f(x_j)$ to Y_{true};
  Append predicted value $\hat{f}_{\text{ens}}(x_j)$ to Y_{pred};
end
Compute c-index between Y_{true} and Y_{pred};
return *c-index*

W In the W strategy (Weighting), all regressors in the ensemble $\mathcal{F}$ contribute to the final prediction. For each test sample, the estimated output is calculated as a weighted average of individual model predictions. The weights are derived from the competence scores $\alpha_n \in \mathcal{A} = \{\alpha_1, \alpha_2, \ldots, \alpha_N\}$, which are normalized according to Equation 15:

$$\tilde{\alpha}_n = \frac{1/\alpha_n}{\sum_{n=1}^{N} 1/\alpha_n} \tag{15}$$

These normalized weights $\tilde{\alpha}_n$ assign greater influence to more competent models (i.e., those with lower α_n). The final prediction for a test instance x_j is then computed using Equation 16:

$$\hat{f}_{\text{ens}}(x_j) = \sum_{n=1}^{N} \tilde{\alpha}_n \cdot \hat{f}_n(x_j) \tag{16}$$

The full procedure is detailed in Algorithm 3.

Algorithm 3: Combining all regressors using W

Input: Ensemble $\mathcal{F}$, Training set $\mathcal{T}'$, Test set $\mathcal{X}$, Weights $\mathcal{W}$, Neighborhood size K
Output: c-index: Concordance Index

Initialize empty lists Y_{true} and Y_{pred};
for *each test pattern* x_j *in* $\mathcal{X}$ **do**
  Find the region of competence Ψ of x_j using $\mathcal{T}'$;
  $\mathcal{A} \leftarrow []$;
  for *each* f_n *in* $\mathcal{F}$ **do**
    Calculate measures $\{m_{1,n}, ..., m_{P,n}\}$ using Ψ;
    $\alpha_n = \sum_{p=1}^{P} w_p \times m_{p,n}$;
    Append α_n to $\mathcal{A}$;
  end
  for *each* α_n *in* $\mathcal{A}$ **do**
    $\tilde{\alpha}_n = \frac{1/\alpha_n}{\sum_{i=1}^{N} 1/\alpha_i}$;
  end
  $\hat{f}_{\text{ens}}(x_j) = \sum_{n=1}^{N} \tilde{\alpha}_n f_n(x_j)$;
  Append true value $f(x_j)$ to Y_{true};
  Append predicted value $\hat{f}_{\text{ens}}(x_j)$ to Y_{pred};
end
Compute c-index between Y_{true} and Y_{pred};
return *c-index*

WS In the WS strategy (Weighting with Selection), each test instance is evaluated by first identifying and removing regressors with competence scores α_n that exceed the midpoint of the score range, i.e., those for which $\alpha_n > (\alpha_{\max} - \alpha_{\min})/2$. This effectively discards the least competent half of the ensemble based on the spread of competence scores. The remaining regressors are then combined to produce the final prediction using Equations 15 and 16. The full procedure is outlined in Algorithm 24.

Algorithm 4: Selecting and Combining the Regressors using WS

Input: Ensemble $\mathcal{F}$; Training set $\mathcal{T}'$; Test set $\mathcal{X}$; Vector of Weights $\mathcal{W}$; Neighborhood size K

Output: *c_index*: Concordance Index

Preds $\leftarrow$ [];
TrueLowerBounds $\leftarrow$ [];
Events $\leftarrow$ [];
for *each test pattern* x_j *in* $\mathcal{X}$ **do**
  Find the region of competence Ψ of x_j using $\mathcal{T}'$;
  for *each* $\hat{f}_n$ *in* $\mathcal{F}$ **do**
    Calculate the measures $\{m_{1,n}, m_{2,n}, ..., m_{P,n}\}$ using Ψ;
    $\alpha_n = \sum_{p=1}^{P} w_p \times m_{p,n}$;
    $\mathcal{A} = \mathcal{A} \cup \alpha_n$;
  $\widetilde{\mathcal{F}} \leftarrow \mathcal{F}$;
  $\widetilde{\mathcal{A}} \leftarrow \mathcal{A}$;
  for *each* $\hat{f}_n$ *in* $\mathcal{F}$ **do**
    if $\alpha_n > (\alpha_{max} - \alpha_{min})/2$ **then**
      $\widetilde{\mathcal{F}} = \widetilde{\mathcal{F}} - \hat{f}_n$;
      $\widetilde{\mathcal{A}} = \widetilde{\mathcal{A}} - \alpha_n$;
  $N = |\widetilde{\mathcal{F}}|$;
  for *each* α_n *in* $\widetilde{\mathcal{A}}$ **do**
    $\tilde{\alpha}_n = \frac{1/\alpha_n}{\sum_{n=1}^{N} 1/\alpha_n}$;
  $\hat{f}_{ens}(x_j) = \sum_{n=1}^{N} \tilde{\alpha}_n \times \hat{f}_n(x_j)$; // $\hat{f}_n \in \widetilde{\mathcal{F}}$
  Append $\hat{f}_{ens}(x_j)$ to Preds;
  Append $f(x_j)$ (lower bound) to TrueLowerBounds;
  Append event status of x_j to Events;
c_index = concordance_index(TrueLowerBounds, Preds, Events);
return *c_index*;

4 Experimental Setup

This section details the experimental setup used to evaluate dynamic ensemble selection (DES) for survival prediction. All experiments were implemented in Python using a modular pipeline to ensure reproducibility across datasets (NMR and NHANES).

4.1 Experiment 1: Model Selection (NMR)

This set of experiments addresses the first two research questions (Q1 and Q2), focusing on model selection and ensemble strategies.

- **Q1** – Does dynamic ensemble selection improve the prediction of kidney graft failure from NMR data compared to static selection and uniform averaging methods?
- **Q2** – Which dynamic ensemble selection strategy—single selection (S), weighting (W), or weighting with selection (WS)—yields the best predictive performance for graft failure on the NMR dataset?

To explore these questions, we construct ensembles of XGBoost models trained on high-dimensional NMR metabolomic data from kidney transplant patients. The goal is to evaluate whether dynamically selecting or weighting models for each test instance leads to improved survival prediction performance.

We compare several prediction strategies, including both static and dynamic approaches. Static ensemble methods involve uniform averaging of all models or clustering-based selection, which apply the same combination rule to all test samples. In contrast, dynamic ensemble selection (DES) methods—**S**, **W**, and **WS**—adaptively select or weight regressors based on local competence measures derived from each test sample's region of competence.

The core hypothesis is that dynamic selection can enhance predictive accuracy by emphasizing models that demonstrate higher local competence in the region of the test sample, thereby reducing the influence of weaker regressors and improving the concordance index (C-index).

To ensure a fair comparison, all methods use identical data preprocessing (including PCA), training procedures, and cross-validation settings. We evaluate each strategy across a range of ensemble sizes, allowing us to analyze performance trends under varying levels of model diversity.

This experimental setup enables a systematic and rigorous comparison of dynamic versus static ensemble strategies, providing insight into their effectiveness for predicting kidney graft failure using high-dimensional NMR data.

4.1.1 Configurations and Hyperparameter Optimization

To optimize model performance for the selection experiments, we conducted hyperparameter tuning via stratified 5-fold cross-validation with a fixed random seed (42) for reproducibility. The optimization targeted XGBoost with Accelerated Failure Time survival objective (XGB-AFT), incorporating early stopping after 50 consecutive boosting rounds without validation improvement.

The resulting optimal hyperparameters are documented in Table 7. This tuning phase aims to establish a reasonably well-performing configuration for fair comparison across ensemble strategies, without claiming global optimality.

Parameter	Value
Learning rate	0.03
Maximum tree depth	2
Minimum child weight	1
AFT loss distribution	Normal
AFT loss distribution scale	0.1
Growth policy	Loss-guided
λ	0.01
α	0.02

Table 3: Best hyperparameters for the XGBoost AFT model obtained from grid search on the NMR dataset.

4.2 Data and Preprocessing

We used an NMR dataset that contains high-dimensional metabolomic profiles annotated with graft survival times and event indicators. The dataset was pre-split into training and external test sets. Principal Component Analysis (PCA) reduced the original 19,000-dimensional feature space to 24 components while preserving 99% of the variance (Figure 6). Clinical covariates such as `Sex` and `Age` were excluded from the PCA transformation and added later.

4.3 Survival Modeling with XGBoost

We constructed ensembles of XGBoost models using the AFT (Accelerated Failure Time) objective with normal loss distribution. The ensemble size varied from 1 to 100 in steps (e.g., 1, 5, 10, ..., 100). Each model was trained on a bootstrapped subset of the training data (`subset_fraction` = 0.7), with 30% reserved for validation during early stopping. Concordance index (C-index) was used to evaluate predictive performance.

4.4 Experiment 2: Genetic Algorithm Optimization

This section presents the second experiment, focused on optimizing dynamic ensemble selection through genetic algorithms (GA).

- **Q3.** Does optimizing DES weights with a genetic algorithm improve graft failure prediction from NMR data?

In this experiment, we use a genetic algorithm (GA) to optimize the competence weights applied in DES strategies. The objective is to improve the concordance index (C-index) by adaptively learning which error measures are most informative for selecting or weighting regressors during inference.

For all replications, the genetic algorithm was configured as follows:

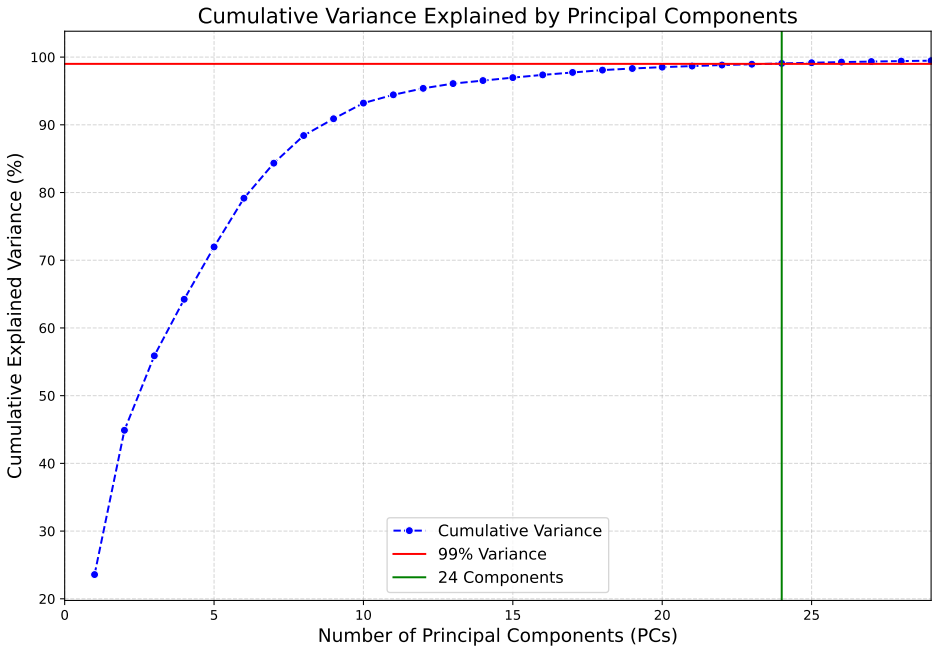

Figure 6: Cumulative explained variance plot from PCA applied to the NMR metabolomic dataset. The first 24 components retain over 99% of the variance, and are used for training and evaluation.

- **Population Size:** 60
- **Initial Mutation Rate:** 0.5
- **Final Mutation Rate:** 0.05
- **Maximum Generations:** 600
- **Stall Generation Limit:** 15 (early stopping if no improvement)
- **Subset Fraction:** 0.4 (fraction of validation data used per generation)
- **Elitism:** Top 6 individuals from each generation are carried forward unchanged
- **Initial Population:**
  - 54 individuals are randomly initialized with real values in the interval $[0, 1]$.
  - 1 chromosome is initialized with all genes set to 1.
  - The remaining chromosomes follow a one-hot initialization, where each chromosome has a 1 in only one gene position and 0 elsewhere, see Matrix 17. This encourages early exploration of isolated error measures.

The GA operates on a fitness function based on the concordance index (C-index) computed over a validation subset. By evolving competence weight vectors over generations, the algorithm learns optimal combinations of local error measures for dynamic selection. The optimized weights are

then evaluated on a held-out test set to assess improvements in predictive performance compared to unoptimized DES configurations.

$$firstPop = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \tag{17}$$

4.5 Additional Dataset: NHANES (Cross–Dataset Validation)

To assess generalizability beyond high-dimensional metabolomics, we applied the same selection framework to the public NHANES cohort with NCHS linked mortality follow-up (time-to-all-cause mortality; right-censoring at last follow-up). Individuals without valid follow-up were excluded.

4.6 Additional Analyses

Beyond discrimination metrics, we prespecified two descriptive analyses to assess calibration and visualize case-level behavior on the external test set. These analyses are purely exploratory and were not used for model selection or hyperparameter tuning.

We plotted the predicted time $\hat{t}_i$ (y-axis) against the observed time t_i (x-axis) for all test subjects, colouring points by outcome (event = red, censored = green). The 45° line ($y{=}x$) serves as a visual reference for perfect pointwise agreement. No recalibration was applied; the plot is purely descriptive.

To illustrate individual behaviour, we randomly sampled five test subjects (using a fixed seed for reproducibility) and drew per-patient timelines. These show the predicted time $\hat{t}_i$ (blue dot), the censoring time where applicable (green square), and the observed event time for failures (orange cross). All times are reported in years since baseline.

5 Results

5.1 Model Selection

This section compares the performance of dynamic ensemble selection (DES) strategies (DES-S, DES-W, DES-WS) against static selection and baseline (uniform averaging) methods. The baseline approach averages predictions across all models, while the static selection employs k-means clustering to identify representative models.

Figure 7 illustrates the mean concordance index (C-index) scores obtained from 5-fold stratified cross-validation across various strategies and ensemble sizes. Overall, the baseline and static methods demonstrate better performance compared to DES strategies. Notably, the static method achieves the highest cross-validation performance at an ensemble size of 50, with a C-index score of 0.74. The relatively poor performance of DES strategies during cross-validation may be attributed to their dependency on accurately identifying local neighborhoods within the training data. Given the limited dataset size, the local neighborhoods may not contain sufficient representative samples, adversely affecting DES performance. This pattern is further emphasized in Figure 8, which displays the top 10 configurations based on cross-validation results. Here, static and baseline methods dominate, with only one DES strategy (DES-WS at ensemble size N=15) appearing among the top configurations.

However, examining the external test set performance in Figure 9, DES strategies (particularly DES-S and DES-WS) display substantially improved results. The availability of the entire training dataset for model selection significantly improves the ability of DES methods to accurately match test instance profiles with suitable models. Despite the relatively modest increase of approximately 40 additional samples compared to cross-validation, the DES methods show marked improvements in predictive accuracy. The DES-WS strategy consistently outperforms static selection and baseline methods, especially at intermediate and larger ensemble sizes (N=40, 60, 80, 100), reaching its highest performance at an ensemble size of 60 with a C-index of 0.75. Interestingly, while the static method excelled during cross-validation, it demonstrates considerable performance instability and ranks among the lowest-performing methods on the external test set, especially after ensemble sizes greater than 50.

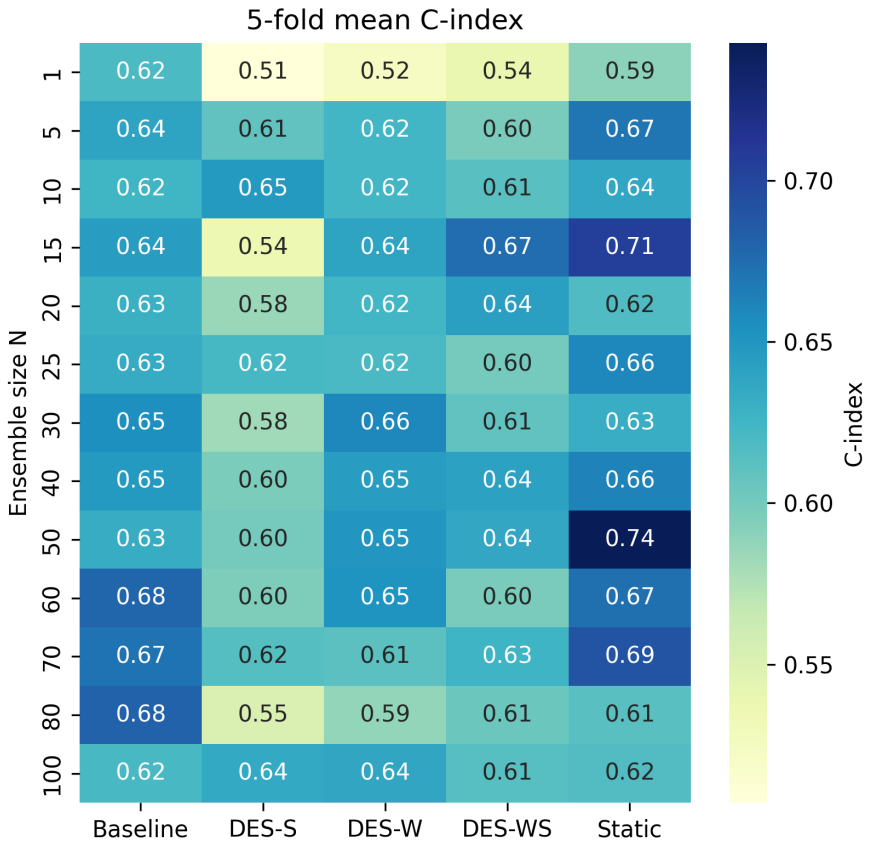

Figure 7: 5-fold stratified cross-validation mean C-index scores for each strategy and ensemble size.

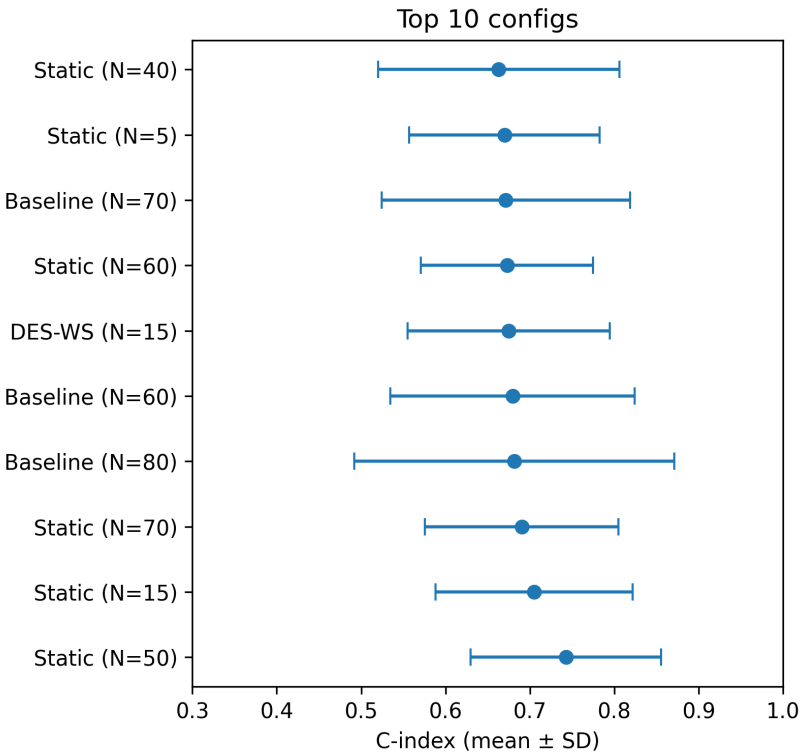

Figure 8: Top 10 configurations based on mean C-index from cross-validation folds.

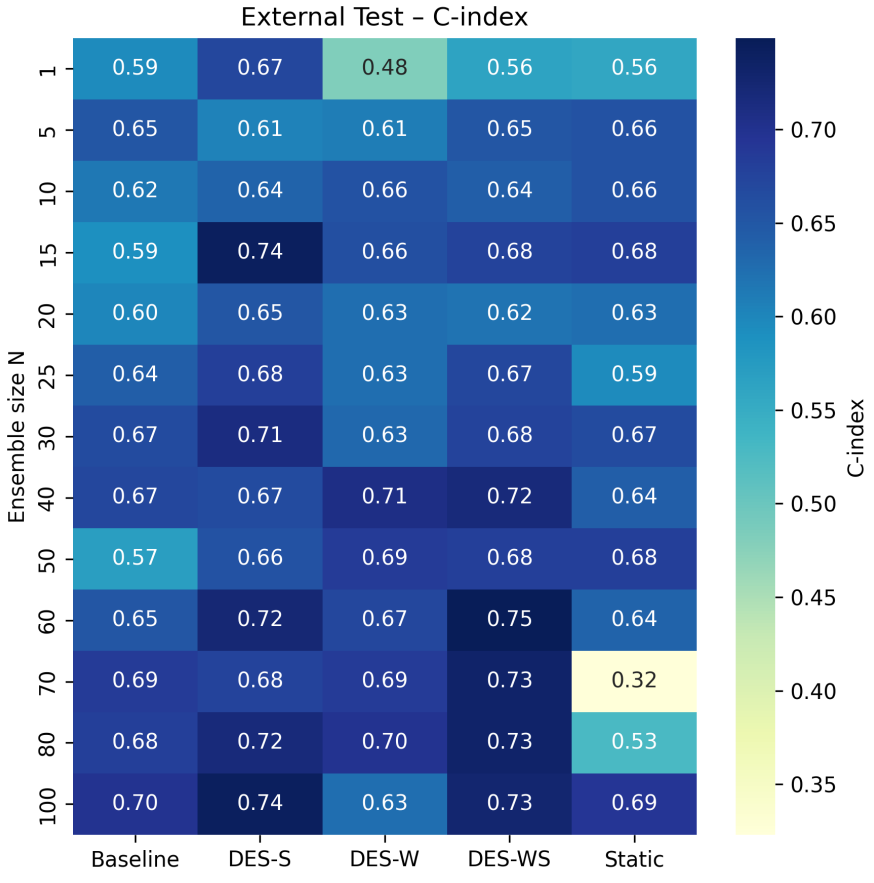

Figure 9: Heatmap of external test set C-index scores by strategy and ensemble size.

5.2 Comparison among DES Strategies

We further investigate the relative performance of the three dynamic ensemble selection (DES) strategies: DES-S, DES-W, and DES-WS.

Figures 10, 11, and 12 compare the performance of the external test set of each strategy; see chapter 3.3.3. Each DES method is defined as follows:

- **DES-S**: Selects the single most competent regressor with the lowest competence score (α_n).
- **DES-W**: Combines all regressors in the ensemble, weighting their predictions inversely proportional to their competence scores.
- **DES-WS**: Selects a subset of top$k$ regressors based on competence scores and combines their predictions using weighted averaging.

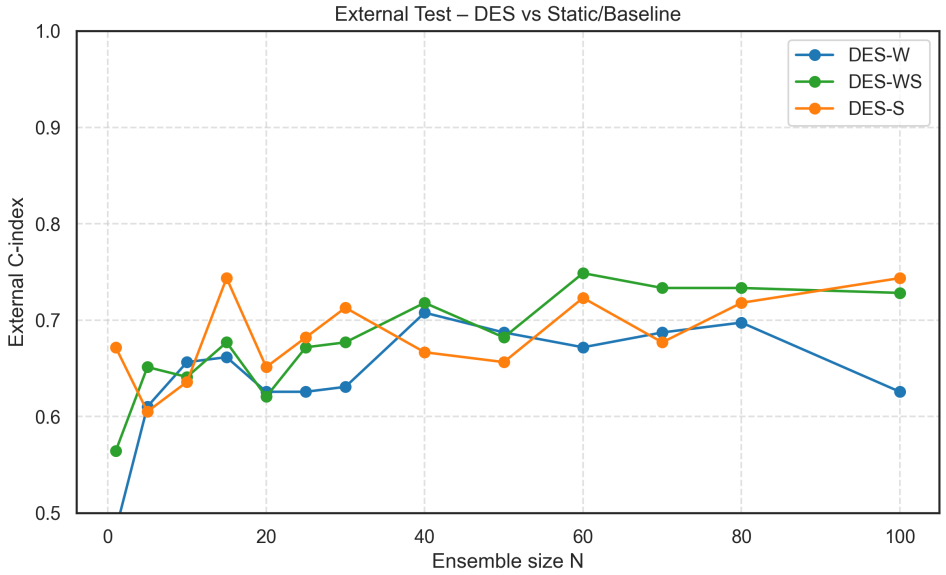

Figure 10: Comparison among DES strategies on the external test set.

In Figure 10, the C-index (y axis) is plotted against ensemble size (x axis). DES-S reaches its maximum performance at $N = 15$ (C-index ≈ 0.74) but shows variability across ensemble sizes. DES-W displays more stable performance across different ensemble sizes, with its peak score at $N = 40$ (C-index ≈ 0.71).

However, the DES-WS strategy demonstrates both robustness and good predictive performance. It consistently improves with ensemble size and peaks at N=60 with a C-index of 0.75, outperforming the other strategies. By combining the most competent subset of models, DES-WS balances performance and stability. This is further supported by the heatmap in Figure 9, which shows that DES-WS frequently achieves the highest scores in all sizes of the ensemble.

This trend is also reflected in Figure 11, which highlights the top 10 best performing configurations on the external test set. DES-WS and DES-S dominate the rankings, with multiple entries each, confirming their competitiveness relative to DES-W and other methods.

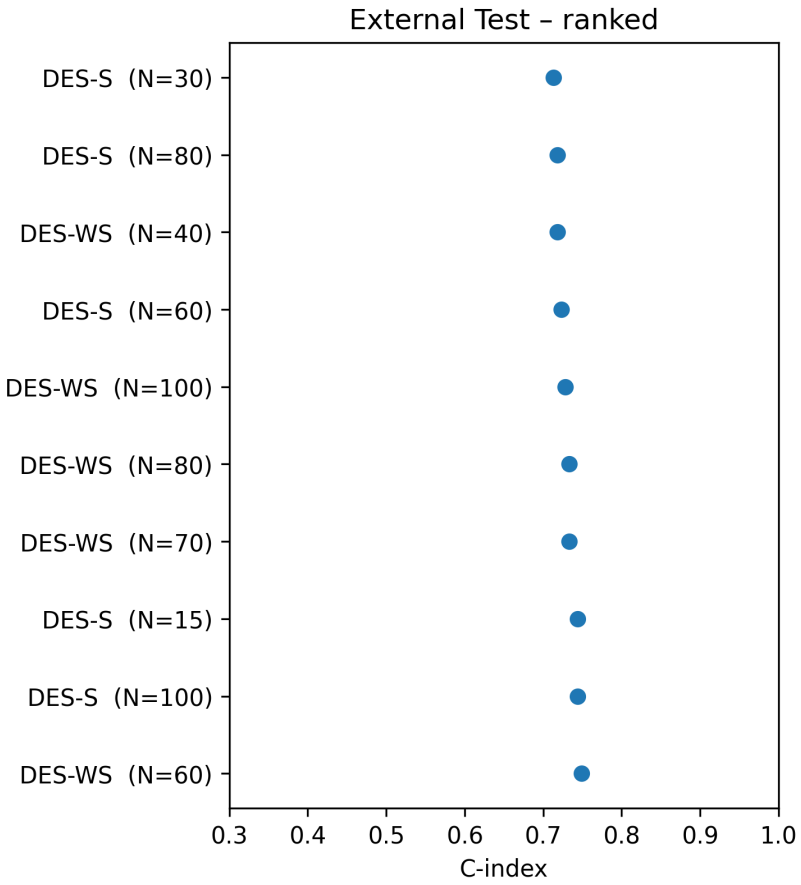

Figure 11: Top 10 configurations based on mean C-index from the external test set.

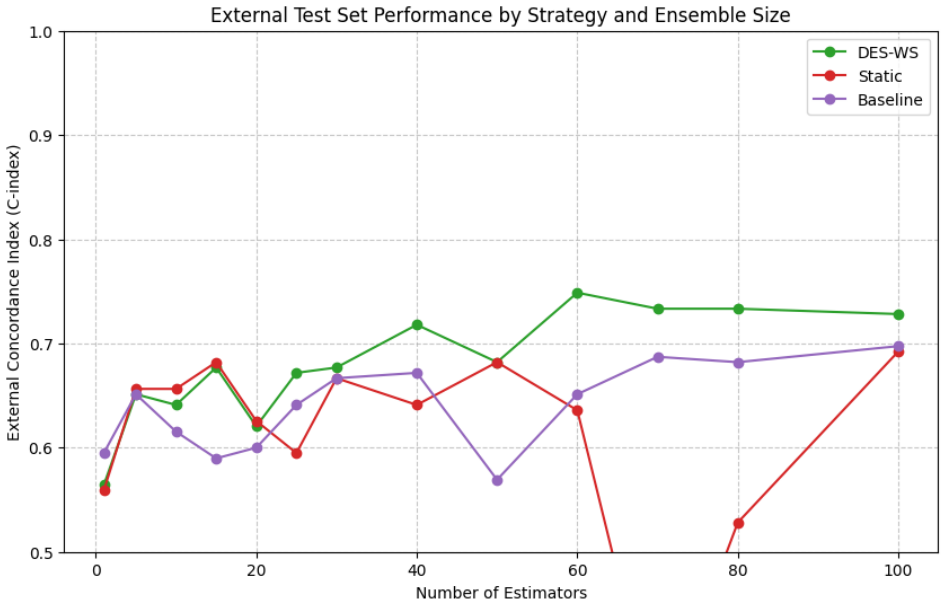

Figure 12: External test set performance (C-index) comparing DES-WS, static, and baseline across ensemble sizes.

Finally, Figure 12 compares the DES-WS strategy to static and baseline methods. For small ensembles ($N < 20$), the static method slightly outperforms DES-WS. However, as the size of the ensemble increases, DES-WS clearly exceeds both baselines, consistently exceeding a C index of 0.7 and peaking at 0.75. In contrast, both static and baseline methods plateau below 0.7 and exhibit higher variance.

These results confirm that DES-WS is the best performing strategy for the NMR dataset. Its ability to balance model competence with ensemble diversity leads to both higher accuracy and better generalization. Given its stable behavior across ensemble sizes and test performance, DES-WS appears particularly well suited for clinical survival prediction tasks where robust, individualized predictions are critical.

However, it is important to note that all DES strategies rely on an accurate estimation of local competence. Since this process depends on the validation set and nearest neighbors (kNN), performance can degrade when the training data are small or unbalanced, as discussed in further detail in Section 4.4.

Table 4: Summary of DES strategy performance (external test set).

Strategy	**Peak C-index**	**Best Ensemble Size (N)**	**Stability**
DES-S	0.74	15	Low (fluctuating)
DES-W	0.71	40	High (stable, lower)
DES-WS	**0.75**	**60**	**High (stable)**

5.3 Impact of Genetic Algorithm Optimization

We examine the impact of genetic algorithm (GA) optimization on competence weight learning within the dynamic ensemble selection (DES) framework. The GA procedure (described in Section 4.4) was designed to find optimal weights to combine multiple competence measures, with the goal of maximizing the concordance index (C index) in the validation set.

Figure 13 compares the performance of the external test set of each DES strategy with and without optimization of the genetic algorithm. While a few configurations benefited from the optimization, the majority of optimized models underperformed relative to their non-optimized counterparts. This performance drop is likely due to overfitting to the small validation set used during optimization, which contained only about 50 samples with five observed events. The limited number of events made it difficult for the algorithm to generalize well, especially for local competence estimation based on the k-nearest neighbors (kNN).

Furthermore, the 5-fold stratified cross-validation setup used for optimization further reduced the number of samples available for reliable neighborhood formation. Since DES strategies are highly dependent on local competence estimation, the resulting neighborhoods were often too small or unrepresentative, weakening the optimization process and leading to sub-optimal weights.

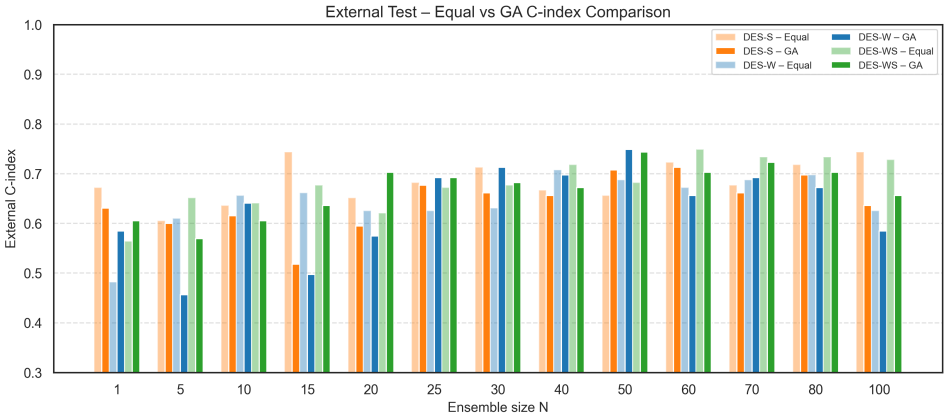

Figure 13: External test set performance (C-index) comparing DES strategies with and without optimized weights. Lighter bars represent equal (non-optimized) weighting; darker bars indicate GA-optimized weights.

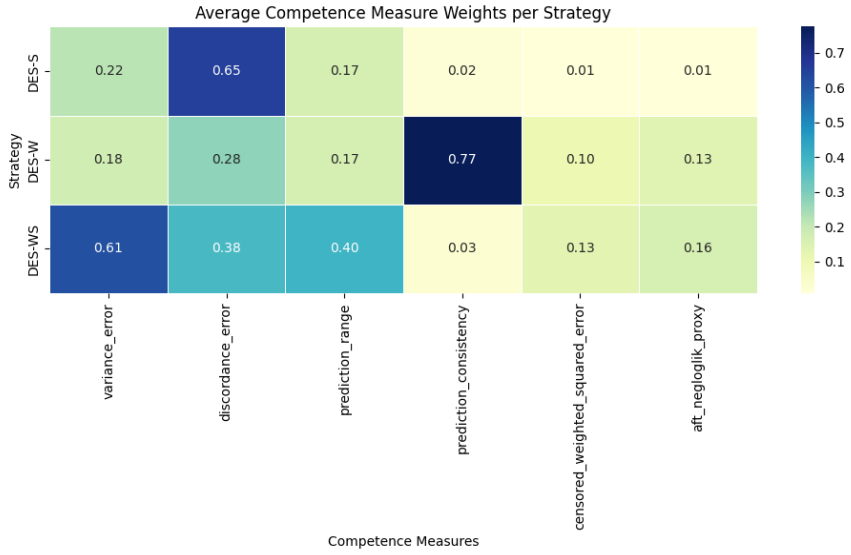

Figure 14: Average optimized competence measure weights per DES strategy (genetic algorithm).

Figure 14 presents the average weights assigned to each competence measure in all strategies after GA optimization. In particular, DES-S placed a heavy emphasis on the discordance error (average weight ≈ 0.65), suggesting that this strategy relied on identifying models with low rank-based disagreement. This could indicate a tendency toward selecting models that produce rankings similar to ground-truth survival times; however, this singular focus may have limited generalization.

DES-W, on the contrary, relied primarily on prediction consistency, assigning it the highest average weight. This aligns with its objective of leveraging the full ensemble while adjusting contributions based on the reliability of individual model predictions across local samples.

DES-WS demonstrated a more balanced weighting across all competence measures, indicating that the combination of selection and weighting benefits from diverse error signals. This diversity could explain its comparatively robust performance, as shown in previous results.

In summary, while genetic algorithm optimization showed potential in tuning competency weights, its effectiveness was limited by data scarcity and local sample instability during validation. Future work may explore hybrid optimization strategies or regularization methods to mitigate overfitting in low-event settings.

5.4 Cross–Dataset Results: NHANES

In NHANES, all the strategies performed very similarly. Across ensemble sizes ($N \in \{1, 5, 10, 15, 20, 30, 50, 100\}$) the held-out C-index clustered tightly around 0.80–0.82 for baseline averaging, static selection, DES-W, and DES-WS, while DES-S was consistently lower by about 0.01–0.03.

Figure 15 reports the 5-fold cross-validation means per N and strategy. For almost all settings, baseline, static, DES-W, and DES-WS yield C-index ≈ 0.81; DES-S is uniformly smaller (≈ 0.79–0.80).

External test results in Figure 16 confirm this pattern: baseline, DES-W, and DES-WS are essentially indistinguishable (≈ 0.81–0.82), static remains close (≈ 0.81), and DES-S trails slightly (≈ 0.79–0.80).

For the three DES variants, Figure 17 shows the external C-index as a function of N. DES-WS and DES-W overlap around 0.81–0.82 with minimal sensitivity to N, while DES-S remains lower and slightly more variable.

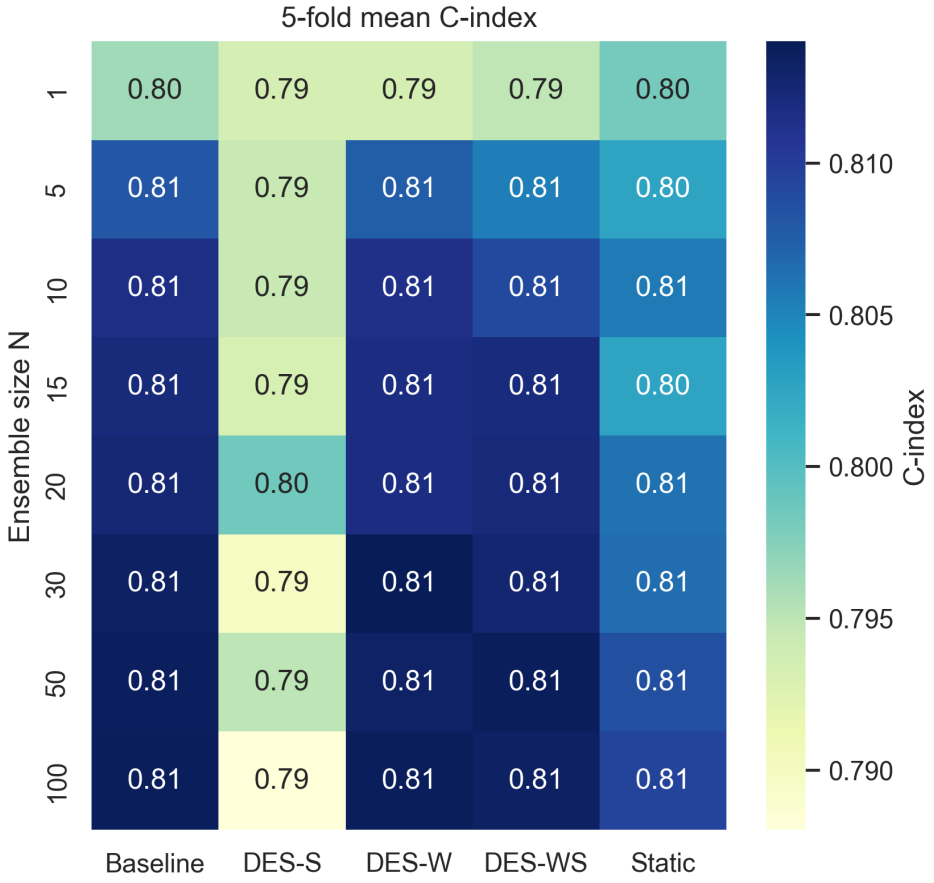

Figure 15: NHANES: 5-fold mean C-index by strategy and ensemble size N. Values concentrate near 0.81 for baseline, static, DES-W, and DES-WS; DES-S is consistently lower.

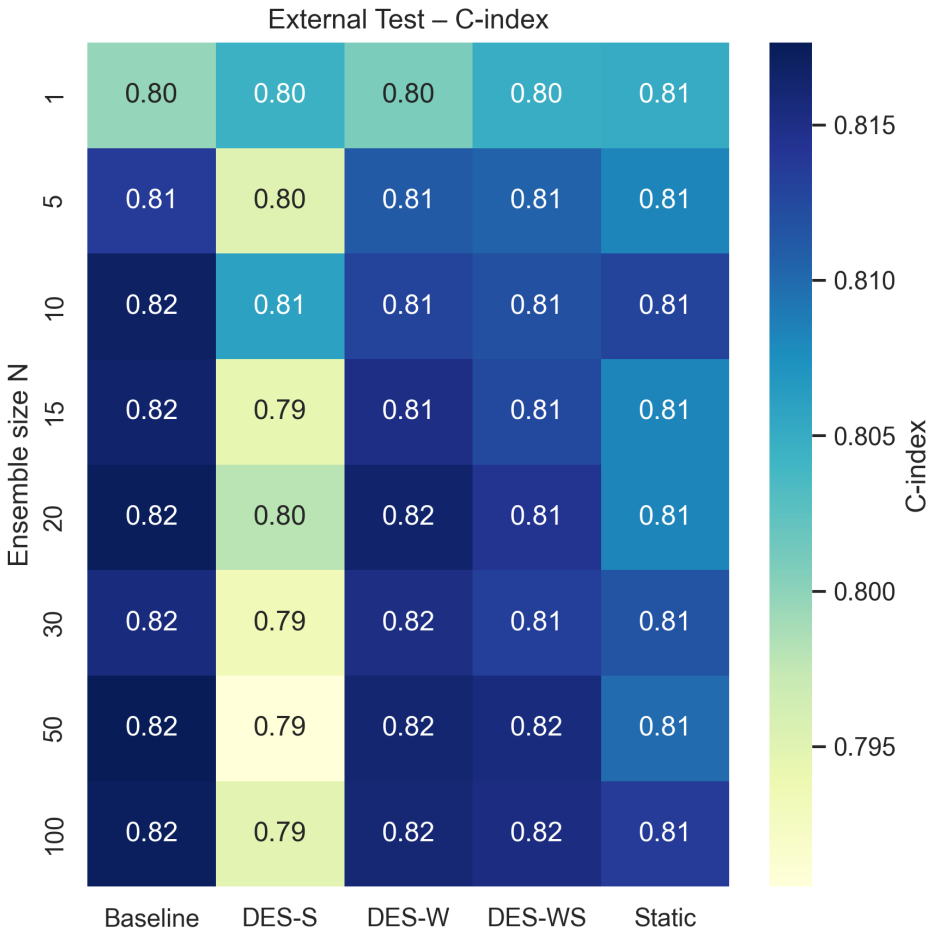

Figure 16: NHANES: external test C-index by strategy and ensemble size N. Baseline, DES-W, and DES-WS are nearly identical (≈ 0.81–0.82); static is comparable; DES-S is slightly lower.

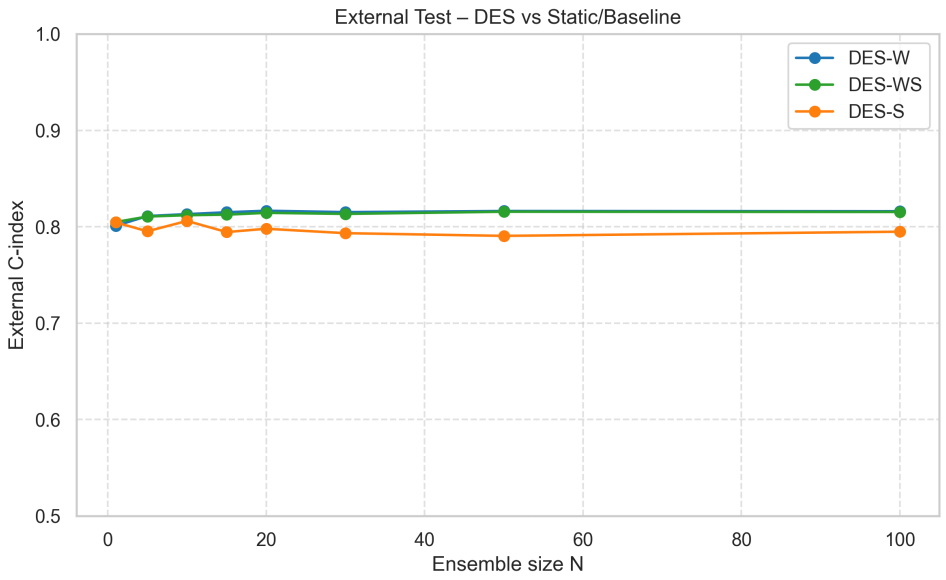

Figure 17: NHANES: external C-index vs. ensemble size N for DES variants. DES-WS and DES-W overlap around 0.81–0.82; DES-S remains $\approx$ 0.79–0.80.

5.5 Additional Analyses

To complement the quantitative evaluation of the ensemble strategies, we conducted additional analyzes to explore model calibration and the interpretability of individual predictions.

Figure 18 presents a calibration graph for the best-performing strategy -DES-WS with $N = 60$ models. Each point represents a single patient: red points indicate observed graft failures, while green points represent censored observations. The diagonal line corresponds to perfect calibration, where the predicted survival time exactly matches the observed or last known time.

The model generally performs well in the midterm survival range (5-8 years), where most red points cluster tightly around the diagonal, suggesting accurate calibration in this critical window. However, for early failures (before year 4), red points are frequently positioned above the diagonal, indicating that the model tends to overestimate survival time when the risk of failure is highest. This behavior reflects a tendency to underestimate the risk in the early posttransplant stages. In contrast, green (censored) cases tend to lie on or above the diagonal, suggesting that the model conservatively predicts shorter survival than actually observed, an encouraging sign from a clinical risk management perspective.

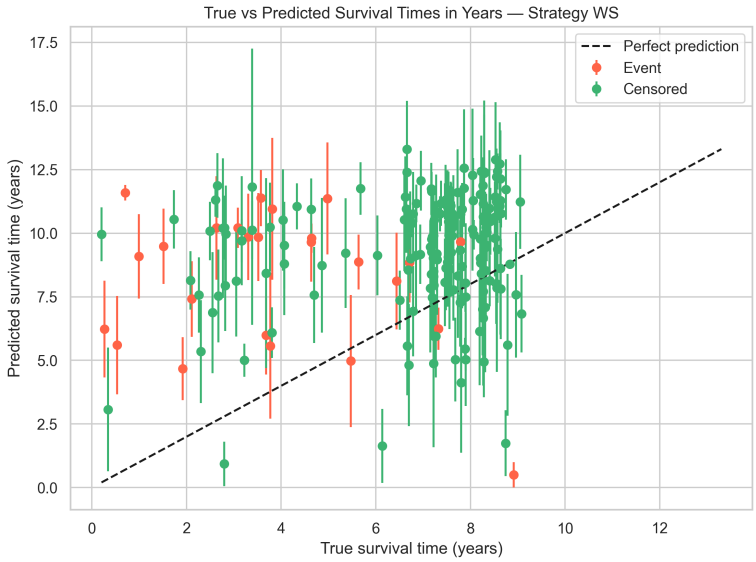

Figure 18: Calibration plot for DES-WS with $N = 60$. Each point represents a patient. Red = graft failure; Green = censored. The diagonal represents perfect calibration (predicted = observed time).

To better illustrate the behavior of the model on an individual level, we also plotted the predicted and observed survival times of five randomly selected patients in Figure 19. Each timeline shows the predicted failure time (blue dot), the time of censoring if applicable (green square), and the observed graft failure (orange cross) if the event occurred.

These examples highlight a variety of behavior patterns in the model. Patient 1 was censored around year 8, with a predicted survival of approximately 10 years. Patient 2, also censored at year 8, had a longer predicted survival of 14 years - both cases are slightly optimistic, but clinically safe. Patient 3 experienced a graft failure in year 7, which the model predicted with high precision (7.2 years). Patient 4 failed early at year 1, but the model predicted 3.5 years, showing an overestimation and a potential risk underestimation in high-risk cases. Lastly, patient 5 was censored at 7.5 years and had a predicted survival of around 11 years, again slightly optimistic but reasonable given the lack of observed failure.

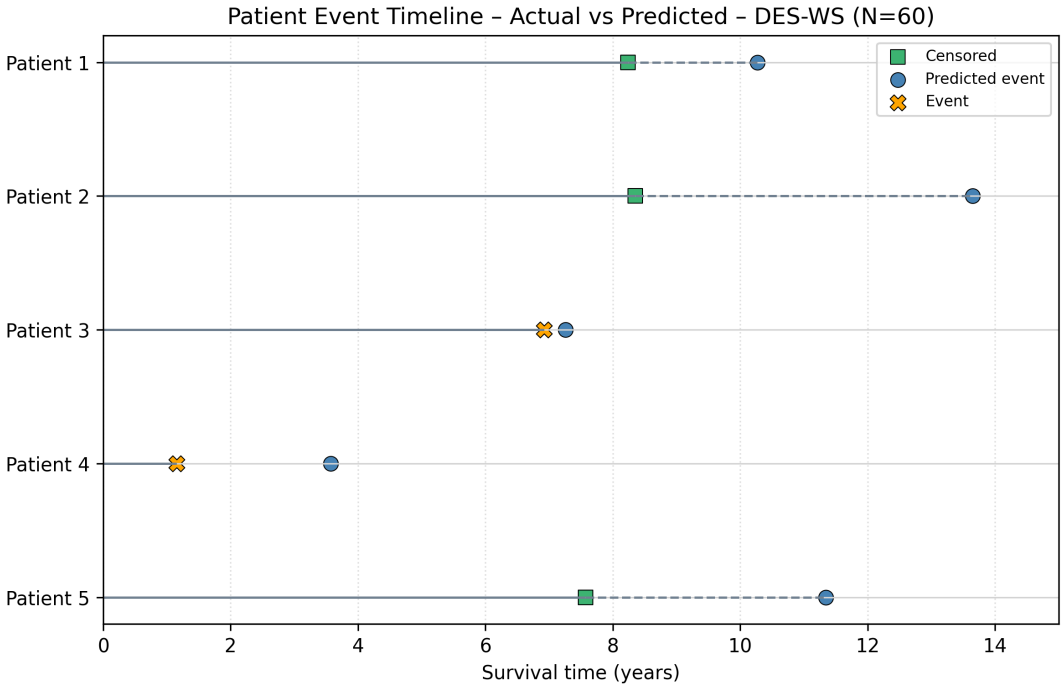

Figure 19: Individual survival time predictions for five randomly selected patients using DES-WS ($N = 60$). Blue dots represent predicted survival times, green squares denote censoring, and orange crosses indicate observed graft failure.

Together, these visual analyses strengthen the findings presented earlier. Although the DES-WS strategy demonstrates strong overall performance, particularly in the mid-term prediction window, there are still limitations in high-risk early failures. However, the model shows reasonable conservatism for censored cases and produces interpretable patient-specific predictions - an important feature for potential clinical integration.

6 Discussion

This study evaluated dynamic ensemble selection (DES) for survival prediction in two settings: a high-dimensional NMR metabolomics cohort for graft-failure risk and a lower-dimensional, population-scale NHANES cohort for time-to–all-cause mortality. We asked whether DES improves upon static selection and uniform averaging, how DES variants compare, and whether genetic-algorithm (GA) weighting adds value. We summarize the findings by question and then note limitations and implications.

6.1 Dynamic Selection vs. Static and Baseline Strategies

NMR (Graft Failure). On the external test set, DES strategies—particularly DES-WS—outperformed both baseline averaging and static selection once the ensemble size exceeded $N > 40$ members, peaking around $N = 60$ (C-index ≈ 0.75). While static selection performed competitively during cross-validation, its performance degraded on the external set. This suggests that selecting a fixed subensemble without access to instance-specific information at test time can limit generalizability.

NHANES (All-Cause Mortality). In contrast, all strategies performed similarly on the NHANES dataset across ensemble sizes ($N \in 1, 5, 10, 15, 20, 30, 50, 100$), with C-indices tightly clustered between 0.80 and 0.82 for the baseline, static, DES-W, and DES-WS methods; DES-S was slightly lower (≈ 0.79–0.80). This lack of separation indicates that when the underlying predictive signal is global and well-captured by the base learners, dynamic instance-level selection offers little advantage over simpler aggregation methods.

These results demonstrate that the benefit of DES is dataset-dependent. DES provides the greatest advantage in high-dimensional, heterogeneous feature spaces (e.g., NMR), where local neighborhoods reveal meaningful instance-specific differences in model competence. When the predictive structure is smoother and base models are in strong agreement (e.g., NHANES), uniform averaging or static selection are sufficient. Thus, our first hypothesis is supported by the NMR data but not by the NHANES data.

6.2 Comparison of DES Strategies

NMR. DES-WS achieved the best balance of accuracy and stability, consistently outperforming DES-W (all-model weighting) and DES-S (single-model selection). DES-S occasionally achieved high accuracy at small N but exhibited higher variance, reflecting the inherent risk of relying on a single model per instance. DES-W was stable but was sometimes diluted by including less competent learners.

NHANES. DES-WS and DES-W were virtually indistinguishable (≈ 0.81–0.82) across all N, while DES-S trailed slightly and was more variable. Consequently, in contexts where DES provides minimal overall benefit (as with NHANES), DES-WS does not harm performance but also offers no clear advantage over DES-W. Overall, our second hypothesis (WS $>$ W $>$ S) holds for the NMR dataset but simplifies to (W $\approx$ WS $>$ S) for NHANES.

6.3 Effect of Genetic Algorithm Optimization

Across both datasets, GA optimization of competence weights failed to produce consistent improvements over equal weighting. On the NMR dataset, the few configurations improved by GA were outweighed by performance decreases on the external set—likely due to overfitting, given the small validation subsets and low event rates used during optimization. On NHANES, where method performance had already converged, GA had a negligible impact. These findings suggest that GA-based weighting may require larger, better-balanced validation sets or stronger regularization with early stopping to prevent overfitting. Therefore, our third hypothesis is not supported.

6.4 Limitations

Several limitations of this study should be acknowledged. (i) The NMR dataset is small with few events, which constrained the stability of both DES neighborhood estimation and GA optimization. (ii) DES performance relies on the definition of k-NN regions of competence; the choice of k, the distance metric, and feature scaling can influence results, and we did not exhaustively retune these hyperparameters for each dataset beyond predefined grids. (iii) Our assessment focused primarily on discrimination (C-index), which measures the ranking of patients by risk but does not quantify the accuracy of predicted event times. Although the supplemental plots suggested reasonable midterm calibration, early failures were overestimated in the NMR data, requiring a more comprehensive calibration analysis for clinical applicability. (iv) The endpoints and data modalities differ (graft failure with metabolomics vs. all-cause mortality with tabular covariates), meaning comparisons between datasets illustrate generalizability rather than direct head-to-head performance.

6.5 Summary and Implications

DES is not universally superior but can be highly advantageous in complex, heterogeneous settings like NMR metabolomics, where no single model is best for all patients. A practical indicator for adopting DES is significant disagreement among base models (e.g., large within-patient spread of predicted times). Where base models agree (e.g., NHANES), simpler aggregation is competitive and easier to deploy.

For translational impact, DES could help surface patient sub-phenotypes of risk that one-size-fits-all models miss—supporting more tailored post-transplant monitoring or therapy adjustments. A pragmatic triage is to adopt DES-WS when exploratory analysis shows strong local heterogeneity or diverse error profiles; otherwise prefer simpler schemes. Future work should (i) pursue more regularized competence optimization, (ii) include survival-specific calibration and decision-curve analyses to gauge clinical utility, and (iii) define simple pre-implementation diagnostics (e.g., model-disagreement scores) to guide when DES is warranted.

7 Conclusion

7.1 Conclusion

This thesis evaluated dynamic ensemble selection (DES) for survival analysis in two complementary settings: predicting kidney graft failure from high-dimensional NMR metabolomics data and forecasting time-to-all-cause mortality in the tabular NHANES cohort. We investigated whether DES outperforms conventional ensembling, which DES variant is most effective, and whether genetic algorithm (GA) optimization of competence weights provides added value.

Three principal conclusions emerge from this work.

(1) The benefit of DES is context-dependent. On the complex, high-dimensional NMR dataset, DES—particularly the weighting-with-selection variant (DES-WS)—surpassed both baseline averaging and static selection on the external test set for ensembles larger than $N > 40$ members, achieving a peak C-index of ≈ 0.75 at $N = 60$. In contrast, all strategies performed similarly on the NHANES dataset (C-index ≈ 0.80–0.82), with DES-S slightly underperforming. This indicates that DES is most advantageous in heterogeneous, high-dimensional feature spaces where model competence varies significantly by instance. In simpler, more homogeneous settings where base learners exhibit strong agreement, conventional aggregation methods are sufficient.

(2) DES-WS provides the most robust performance. DES-WS consistently delivered an optimal balance of stability and accuracy across experiments. It outperformed the high-variance DES-S, which relies on a single model per instance, and often exceeded DES-W, which can be diluted by including less competent models. On NHANES, where the potential for improvement was limited, DES-WS and DES-W performed identically, further confirming the pragmatic advantage of the hybrid approach.

(3) GA optimization failed to enhance performance. GA-derived competence weights did not consistently outperform equal weighting and in some cases reduced external validation performance, likely due to overfitting on small, event-sparse validation sets. This suggests that the instability of local neighborhoods in limited data settings makes the optimization problem particularly challenging.

In summary, this work establishes DES as a powerful but specialized tool. It is most effective in settings with substantial local heterogeneity and diverse base model error profiles (e.g., NMR metabolomics). This approach is particularly suited for clinical problems where patient populations are not uniform and outcomes are driven by multiple biological pathways. In the absence of these conditions (e.g., NHANES), simpler ensemble strategies remain competitive and operationally simpler to implement.

7.2 Future Work

Several directions follow from these findings.

Data scale and external validation. Larger, multi-center transplant cohorts with more events would strengthen DES neighborhood estimation and provide fairer grounds for weight optimization. Prospective external validation is critical for clinical adoption.

Robust competence optimization. Replace GA with more regularized schemes (Bayesian optimization with priors, multi-objective evolution with parsimony pressure, or cross-fitted meta-learners) and add early stopping / shrinkage to reduce overfitting under sparse events.

Adaptive regions of competence. Learn the neighborhood itself: metric learning or representation learning tailored to survival objectives; event-aware distances; variable k that scales with local event density.

Calibration and clinical utility. Extend beyond discrimination to thorough calibration (time-dependent calibration curves, Brier scores) and decision-curve analysis to quantify net benefit across thresholds—especially in early high-risk windows. This is essential for translating predicted risks into actionable clinical decisions.

Interpretability. Augment DES with survival-specific attribution (e.g., SHAP for AFT/XGBoost with temporal summaries) and pathway-level analyses for NMR to make predictions clinically actionable.

Taken together, this work positions DES as a targeted tool for personalized survival prediction: valuable when heterogeneity is substantial, and otherwise gracefully deferring to simpler ensembles. With better diagnostics, larger datasets, and more robust optimization, DES can be deployed where it is most likely to deliver clinical benefit.

References

Barnwal, A., Cho, H., & Hocking, T. (2022, October). Survival Regression with Accelerated Failure Time Model in XGBoost. *Journal of Computational and Graphical Statistics*, *31*(4), 1292–1302. Retrieved 2024-02-23, from https://doi.org/10.1080/10618600.2022.2067548 (Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/10618600.2022.2067548) doi: 10.1080/10618600.2022.2067548

Basics of survival analysis in biostatistics. (n.d.). https://www.linkedin.com/pulse/basics-survival-analysis-biostatistics-jesca-birungi--cleze.

Breiman, L. (2001). Random forests. *Machine learning*, *45*, 5–32.

Caruana, R., Niculescu-Mizil, A., Crew, G., & Ksikes, A. (2004). Ensemble selection from libraries of models. In *Twenty-first international conference on Machine learning - ICML '04* (p. 18). Banff, Alberta, Canada: ACM Press. Retrieved 2024-03-01, from http://portal.acm.org/citation.cfm?doid=1015330.1015432 doi: 10.1145/1015330.1015432

Centers for Disease Control and Prevention. (n.d.). *National center for health statistics (nchs).* Retrieved 2025-08-22, from https://www.cdc.gov/nchs/

Chen, T., & Guestrin, C. (2016, August). XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 785–794). Retrieved 2024-02-08, from http://arxiv.org/abs/1603.02754 (arXiv:1603.02754 [cs]) doi: 10.1145/2939672.2939785

Chen, Y., Jia, Z., Mercola, D., & Xie, X. (2013). A gradient boosting algorithm for survival analysis via direct optimization of concordance index. *Computational and mathematical methods in medicine*, *2013*(1), 873595.

Cobas, C. (2020). Nmr signal processing, prediction, and structure verification with machine learning techniques. *Magnetic Resonance in Chemistry*, *58*(6), 512–519.

Coelho, G. P., & Von Zuben, F. J. (2006). Aria: An adaptive radius immune algorithm.

Corsaro, C., Vasi, S., Neri, F., Mezzasalma, A. M., Neri, G., & Fazio, E. (2022). Nmr in metabolomics: From conventional statistics to machine learning and neural network approaches. *Applied Sciences*, *12*(6), 2824.

Costanti, F., Kola, A., Scarselli, F., Valensin, D., & Bianchini, M. (2023). A deep learning approach to analyze nmr spectra of sh-sy5y cells for alzheimer's disease diagnosis. *Mathematics*, *11*(12), 2664.

Cox, D. R. (1972). Regression models and life-tables. *Journal of the Royal Statistical Society: Series B (Methodological)*, *34*(2), 187–202.

Cruz, R. M., Sabourin, R., & Cavalcanti, G. D. (2018). Dynamic classifier selection: Recent advances and perspectives. *Information Fusion*, *41*, 195-216. Retrieved from https://www.sciencedirect.com/science/article/pii/S1566253517304074 doi: https://doi.org/10.1016/j.inffus.2017.09.010

Deelen, J., Kettunen, J., Fischer, K., van Der Spek, A., Trompet, S., Kastenmüller, G., ... others (2019). A metabolic profile of all-cause mortality risk identified in an observational study of 44,168 individuals. *Nature communications*, *10*(1), 3346.

Di Donato, S., Vignoli, A., Biagioni, C., Malorni, L., Mori, E., Tenori, L., ... others (2021). A serum metabolomics classifier derived from elderly patients with metastatic colorectal cancer predicts relapse in the adjuvant setting. *Cancers*, *13*(11), 2762.

Dietterich, T. G. (2000). Ensemble methods in machine learning. In *Multiple classifier systems* (pp. 1–15). Berlin, Heidelberg: Springer Berlin Heidelberg.

Eiben, A. E., & Smith, J. E. (2015). *Introduction to evolutionary computing*. Springer.

Harrell, F. E. (2015). *Regression Modeling Strategies: With Applications to Linear Models, Logistic and Ordinal Regression, and Survival Analysis*. Cham: Springer International Publishing. Retrieved 2024-02-16, from https://link.springer.com/10.1007/978-3-319-19425-7 doi: 10.1007/978-3-319-19425-7

Hu, L., Ji, J., & Li, F. (2021). Estimating heterogeneous survival treatment effect in observational data using machine learning. *Statistics in medicine*, *40*(21), 4691–4713.

Johnson, C. H., Ivanisevic, J., & Siuzdak, G. (2016). Metabolomics: beyond biomarkers and towards mechanisms. *Nature reviews Molecular cell biology*, *17*(7), 451–459.

Kaplan, E. L., & Meier, P. (1958). Nonparametric estimation from incomplete observations. *Journal of the American Statistical Association*, *53*(282), 457–481. Retrieved 2025-06-21, from http://www.jstor.org/stable/2281868

Kaynar, G., Cakmakci, D., Bund, C., Todeschi, J., Namer, I. J., & Cicek, A. E. (2023). Pideel: metabolic pathway-informed deep learning model for survival analysis and pathological classification of gliomas. *Bioinformatics*, *39*(11), btad684.

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., ... Liu, T.-Y. (2017). LightGBM: A Highly Efficient Gradient Boosting Decision Tree. In *Advances in Neural Information Processing Systems* (Vol. 30). Curran Associates, Inc. Retrieved 2024-03-07, from https://proceedings.neurips.cc/paper_files/paper/2017/hash/6449f44a102fde848669bdd9eb6b76fa-Abstract.html

Khan, M. H. R., & Shaw, J. E. H. (2016). Variable selection for survival data with a class of adaptive elastic net techniques. *Statistics and Computing*, *26*(3), 725–741.

Ko, A. H., Sabourin, R., & Britto Jr, A. S. (2008). From dynamic classifier selection to dynamic ensemble selection. *Pattern recognition*, *41*(5), 1718–1731.

Lazarevic, A., & Obradovic, Z. (2001). Effective pruning of neural network classifier ensembles.

Meier-Kriesche, H.-U., Schold, J. D., & Kaplan, B. (2004). Long-term renal allograft survival: have we made significant progress or is it time to rethink our analytic and therapeutic strategies? *American Journal of Transplantation*, *4*(8), 1289–1295.

Meier-Kriesche, H.-U., Schold, J. D., Srinivas, T. R., & Kaplan, B. (2004). Lack of improvement in renal allograft survival despite a marked decrease in acute rejection rates over the most recent era. *American journal of transplantation*, *4*(3), 378–383.

Mendes-Moreira, J., Jorge, A. M., Soares, C., & de Sousa, J. F. (2009). Ensemble learning: A study on different variants of the dynamic selection approach. In *Machine learning and data mining in pattern recognition: 6th international conference, mldm 2009, leipzig, germany, july 23-25, 2009. proceedings 6* (pp. 191–205).

Mendes-Moreira, J., Soares, C., Jorge, A. M., & Sousa, J. F. (2015). Ensemble approaches for regression: A survey. *ACM Computing Surveys*.

Nicholson, J. K., Buckingham, M. J., & Sadler, P. J. (1983). High resolution 1h nmr studies of vertebrate blood and plasma. *Biochemical Journal*, *211*(3), 605–615.

Partalas, I., Tsoumakas, G., Hatzikos, E. V., & Vlahavas, I. (2008). Greedy regression ensemble selection: Theory and an application to water quality prediction. *Information Sciences*, *178*(20), 3867–3879.

Peng, W. K., Ng, T.-T., & Loh, T. P. (2020). Machine learning assistive rapid, label-free molecular

phenotyping of blood with two-dimensional nmr correlational spectroscopy. *Communications biology*, *3*(1), 535.

Pickett, K., Suresh, K., Miller, K., Davis, S., & Juarez, E. (2021, 10). Random survival forests for dynamic predictions of a time-to-event outcome using a longitudinal biomarker. *BMC Medical Research Methodology*, *21*. doi: 10.1186/s12874-021-01375-x

Pölsterl, S. (2020). scikit-survival: A library for time-to-event analysis built on top of scikit-learn. *Journal of Machine Learning Research*, *21*(212), 1-6. Retrieved from http://jmlr.org/papers/v21/20-729.html

Rooney, N., Patterson, D., & Nugent, C. (2004). Dynamic classifier selection using regression. *IDA*.

Ruta, D., & Gabrys, B. (2001). Application of the evolutionary algorithms for classifier selection in multiple classifier systems with majority voting. In *International workshop on multiple classifier systems* (pp. 399–408).

Senanayake, S., White, N., Graves, N., Healy, H., Baboolal, K., & Kularatna, S. (2019, October). Machine learning in predicting graft failure following kidney transplantation: A systematic review of published predictive models. *International Journal of Medical Informatics*, *130*, 103957. Retrieved 2024-03-05, from https://www.sciencedirect.com/science/article/pii/S1386505619302977 doi: 10.1016/j.ijmedinf.2019.103957

Valdovinos, R. M., Sánchez, J. S., & Barandela, R. (2005). Dynamic and static weighting in classifier fusion. In J. S. Marques, N. Pérez de la Blanca, & P. Pina (Eds.), *Pattern recognition and image analysis* (pp. 59–66). Berlin, Heidelberg: Springer Berlin Heidelberg.

van den Berg, E., Pasch, A., Westendorp, W. H., Navis, G., Brink, E. J., Gans, R. O., ... Bakker, S. J. (2014). Urinary sulfur metabolites associate with a favorable cardiovascular risk profile and survival benefit in renal transplant recipients. *Journal of the American Society of Nephrology*, *25*(6), 1303–1312.

Woods, K., Kegelmeyer, W. P., & Bowyer, K. (1997). Combination of multiple classifiers using local accuracy estimates. In *Cvpr*.

Yan, F., & Feng, Y. (2022). A two-stage stacked-based heterogeneous ensemble learning for cancer survival prediction. *Complex & Intelligent Systems*, *8*(6), 4619–4639.

Zhou, Z.-H. (2015). Ensemble learning. In S. Z. Li & A. K. Jain (Eds.), *Encyclopedia of biometrics* (pp. 411–416). Boston, MA: Springer US. Retrieved from https://doi.org/10.1007/978-1-4899-7488-4_293 doi: 10.1007/978-1-4899-7488-4_293

Appendices

A GA optimization results

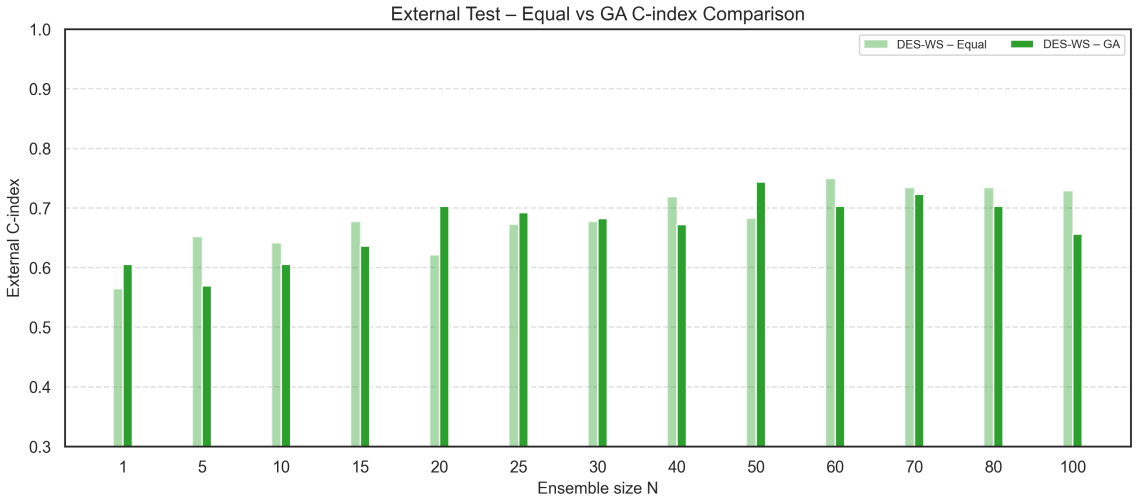

Figure 20: Comparison of external C-index performance between the DES-WS ensemble using equal weighting (light green) and GA-optimized weighting (dark green) across different ensemble sizes N. The results show no improvement for the majority of configurations

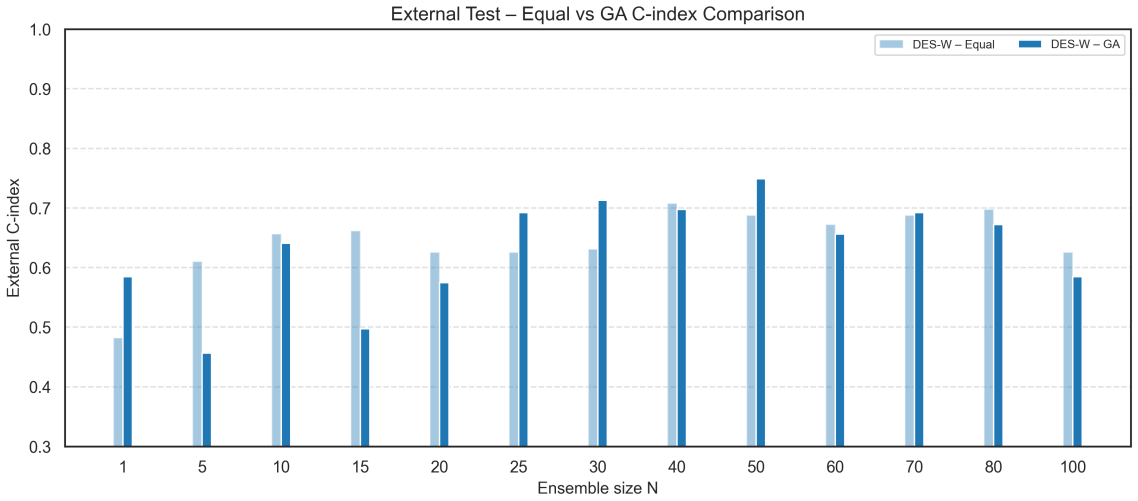

Figure 21: Comparison of external C-index performance between the DES-WS ensemble using equal weighting (light green) and GA-optimized weighting (dark green) across different ensemble sizes N. The results show no improvement for the majority of configurations

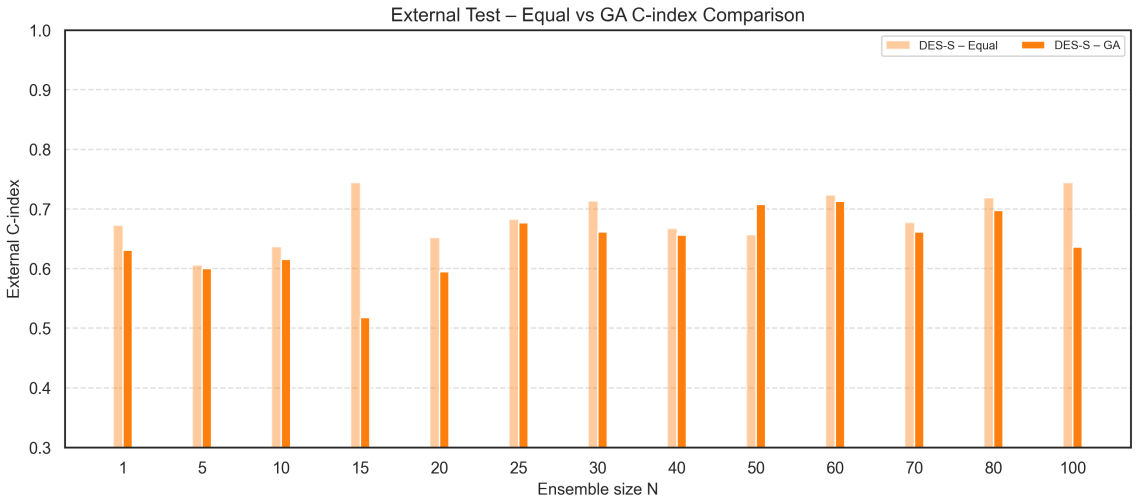

Figure 22: Comparison of external C-index performance for the DES-S ensemble using equal weighting (light orange) versus GA-optimized weighting (dark orange) across different ensemble sizes N. The results show no improvement for the majority of configurations.

B Optimization

C XGBoost Hyperparameter Search Space

Table 5: Tuned hyperparameters for XGBoost (grid search).

Hyperparameter	**Values explored**
Learning rate	{0.01, 0.03, 0.1}
Maximum tree depth	{2, 3, 4, 5, 6, 7, 8}
Minimum child weight	{1, 2, 3, 4, 5}
Alpha (L1 regularization)	{0.01, 0.03, 0.1}
Lambda (L2 regularization)	{0.01, 0.03, 0.1}

Table 6: Fixed hyperparameters for XGBoost.

Hyperparameter	**Value**
Objective	survival:aft
Evaluation metric	aft-nloglik
AFT loss distribution	normal
Tree method	hist
Seed	42
Grow policy	lossguide
AFT loss distribution scale	0.1
Booster	gbtree

Table 7: Final optimized XGBoost hyperparameters for the NMR dataset.

Parameter	**Value**
Objective	survival:aft
Evaluation metric	aft-nloglik
AFT loss distribution	normal
Learning rate	0.03
Max depth	2
Min child weight	1
Alpha (L1 regularization)	0.02
Lambda (L2 regularization)	0.01
Tree method	hist
Seed	42
Grow policy	lossguide
AFT loss distribution scale	0.1
Booster	gbtree