A Joint Model Framework for Longitudinal and Survival Analysis

Likelihood Derivation and Simulation-Based Evaluation of a Shared-Parameter Joint Model

Mark Fulginiti

2025-11-07

Contents

Introduction

Modern biomedical and reliability studies frequently collect two types of information: longitudinal measurements describing how a biological or mechanical system evolves over time, and time-to-event outcomes indicating when a critical transition occurs, such as death, relapse, or equipment failure. These two processes are rarely independent. Subjects who deteriorate more rapidly are also more likely to experience earlier events or to drop out of follow-up, while the occurrence of the event itself halts the collection of further longitudinal data. Traditional statistical approaches often treat these processes separately, using mixed models for the longitudinal responses and survival models for the event times, but this separation can distort inference when the processes are correlated.

Missing or incomplete data further complicate this setting. Longitudinal observations are often measured with error, unequally spaced in time, and truncated by the event process. Event times, in turn, are subject to censoring or truncation due to study design or loss to follow-up. In both processes, the probability of observation may depend on unobserved or outcome-related quantities, making the missingness informative. Marginal analyses that ignore this dependence yield biased estimates. Joint models address these issues by formulating the longitudinal and survival components together within a unified likelihood that accommodates informative missingness and shared latent structure.

The present analysis focuses on joint models that integrate a longitudinal submodel for the internal covariate trajectory with a survival submodel for event timing through shared latent random effects. This structure permits the joint distribution of repeated measures and event times to be expressed via a single likelihood, accommodating informative dropout and event-dependent observation. Among joint model formulations, the transformation model provides particular flexibility by allowing the cumulative hazard to be expressed through a parametric transformation function, encompassing the proportional hazards and proportional odds models as special cases. The following sections outline the separate longitudinal and survival components before combining them into the full joint likelihood.

Longitudinal models are required when dealing with internal covariates. Following Wang and Zhong (2025), a time-dependent covariate W with history $\mathcal{W}(t) = \{W(s), \quad 0 \leq s < t\}$ up to event time $T = t$ is said to be external if, for all u, v such that $0 < u \leq v$, the following property holds:

$$P(u \leq T \leq u + du \mid \mathcal{W}(u), \quad T \geq u) = P(u \leq T \leq u + du \mid \mathcal{W}(v), \quad T \geq u).$$

A covariate that does not satisfy this equality is called internal. In other words, internal covariates are those whose future values are influenced by the event process itself or by latent factors linked with the event under study. Their evolution is not independent of survival time. This dependence means the covariate trajectory and the event risk are jointly determined, and conditioning on their history through their inclusion in the model changes the event probability.

Since internal covariates violate the exogeneity condition required for standard time-dependent survival models, that is, the covariate is correlated with the unobserved components (error terms) of the hazard model, leading to biased estimates, separate modeling of their longitudinal process is necessary. The longitudinal submodel captures how these internal covariates evolve over time by accounting for random effects and measurement error, and their estimated latent trajectories can then be linked to the survival model through shared parameters or current-value association.

By contrast, external covariates, which are those satisfying the equality above, remain independent of the event mechanism and can safely enter the hazard model directly as time-dependent predictors without inducing bias. These external covariates, whether time-dependent or fixed, are assumed to be fully observed and measured without error, whereas internal covariates are inherently time-dependent and only partially observed, making them prone to measurement error and bias if modeled naively.

While longitudinal models describe how internal covariates evolve over time, they do not account for the

timing of events that may terminate observation, such as death, dropout, or relapse, which are often the very events of interest. This is the domain of survival analysis, which focuses on modeling the hazard function,

$$\lambda(t) = \lim_{\Delta t \to 0} \frac{P(t \leq T < t + \Delta t \mid T \geq t)}{\Delta t},$$

which represents the instantaneous risk of experiencing the event at time t, conditional on having survived up to t. Modeling the hazard, rather than the outcome itself, allows one to incorporate censoring (instances where the event time is only partially observed) and to estimate how covariates influence the risk of event occurrence over time. The most common types are right-censoring and interval-censoring, which occur when the exact event time is unknown but bounded. In right-censoring, the event has not yet occurred by the last observation time, so we only know that $T_i > C_i$, where C_i is the censoring time. In interval-censoring, the event occurs between two observation times, such that $L_i < T_i \leq R_i$. A related case is left-censoring, which arises when an individual enters the study after the biological or mechanical process of interest has already begun, so the only information available is that the event occurred before a known time point; formally, $T_i \leq L_i$. All three situations yield only partial information about T_i, requiring likelihood formulations that integrate over the unobserved portions of the event-time distribution.

A related issue is truncation, which limits who enters the study in the first place. In left truncation, individuals whose event times occur before study entry are unobserved and the subject is included in the sample only if $T_i > L_i$. In right truncation, only subjects who have already experienced the event by a cutoff time are included ($T_i \leq R_i$). Truncation alters the sampling distribution of event times and, if not properly accounted for, can bias inference in the same way as censoring.

Left censoring and left truncation describe different kinds of partial information. Under left censoring, the subject is in the study, but the exact event time is known only to satisfy $T_i \leq L_i$; the process has already begun, but the event has not been directly observed. Under left truncation, by contrast, subjects whose event times occur before study entry are entirely unobserved. A subject is included only if $T_i > L_i$. Thus, left truncation modifies the sampling distribution by excluding early failures altogether, while left censoring retains such individuals but with incomplete information. Both mechanisms can occur in the same study, e.g., subjects may enter after the onset of risk (introducing left censoring), and individuals who experienced the event before entry are never observed (introducing left truncation). When either mechanism is ignored, the resulting likelihood no longer represents the appropriate conditional distribution, leading to biased estimation.

When covariates are internal, their values evolve in a manner that depends on the same latent factors driving the hazard process. The longitudinal and survival processes are therefore dependent, and separate modeling of each induces bias: the longitudinal model fails to account for event-driven truncation and informative missingness, while the survival model treats the internal covariate as error-free and ignores uncertainty in its latent trajectory.

The goal of this analysis is to derive and evaluate the likelihood-based estimators for a transformation joint model through simulation under varying degrees of association and measurement error.

Models

Linear mixed model for longitudinal data

The linear mixed model (LMM) for repeated measures decomposes variability into fixed and random components while capturing both within-subject (serial) and between-subject (cross-sectional) correlation (Laird & Ware, 1982; Cooper et al., 2013). Fixed effects represent population-average parameters, describing the expected influence of covariates on the outcome across all subjects. Random effects, denoted b_i, are subject- or cluster-specific parameters that capture deviations from the population mean structure defined by the fixed effects, thereby modeling heterogeneity across individuals or clusters. In short, the fixed effects

determine the mean structure of the population, while the random effects define the covariance structure, capturing the correlation among repeated measures within subjects.

For subject $i = 1, \ldots, n$ observed at times t_{ij}, the longitudinal measurements follow

$$Y_{ij} = x_{ij}^\top \beta + z_{ij}^\top b_i + \varepsilon_{ij}, \quad b_i \sim N(0, \Sigma_b), \quad \varepsilon_{ij} \sim N(0, R_i), \quad b_i \perp \varepsilon_{ij},$$

where Y_{ij} denotes the response recorded at time t_{ij}. The vector x_{ij} contains the fixed-effect covariates describing the population mean structure, and β is the corresponding vector of fixed-effect parameters. The vector z_{ij} contains the random-effect covariates, and b_i collects the subject-specific deviations from the fixed effects. The random effects follow a multivariate normal distribution with covariance matrix Σ_b; the element $(\Sigma_b)_{pq} = \mathrm{Cov}(b_{ip}, b_{iq})$ gives the covariance between the p-th and q-th random-effect components. Diagonal entries represent variances of individual random effects (such as a random intercept or slope), while off-diagonal entries represent covariances between them. For example, a random intercept α_i and random slope β_i can be written as

$$b_i = \begin{pmatrix} \alpha_i \\ \beta_i \end{pmatrix}, \quad \Sigma_b = \begin{pmatrix} \sigma_\alpha^2 & \sigma_{\alpha\beta} \\ \sigma_{\alpha\beta} & \sigma_\beta^2 \end{pmatrix},$$

where $\sigma_{\alpha\beta}$ quantifies the correlation between individual intercepts and slopes.

The residual term ε_{ij} captures within-subject variability, and its covariance structure is encoded in the matrix R_i. The entry $(R_i)_{jk} = \mathrm{Cov}(\varepsilon_{ij}, \varepsilon_{ik})$ gives the residual covariance between observations at times t_{ij} and t_{ik}. When $j = k$, this entry represents the residual variance at a single time point; when $j \neq k$, it represents the serial correlation between repeated measurements. Different choices for R_i impose different assumptions on this correlation. Independent errors correspond to $R_i = I$, implying no residual correlation across time. Compound symmetry assumes a constant correlation ρ among all residual pairs, with off-diagonal elements equal to $\rho\sigma^2$. An autoregressive structure of order one (AR(1)) allows correlation to decay with the temporal lag, with $(R_i)_{jk} = \sigma^2 \rho^{|t_{ij} - t_{ik}|}$. More flexible forms, including unstructured, Toeplitz, moving-average, and spatial covariance matrices, may be used when the sampling design or scientific context suggests more complex dependence patterns.

Within-subject vs between-subject correlation

Repeated observations on an individual are correlated because of shared random effects and possibly serially correlated residuals:

$$Cov(Y_{ij}, Y_{ik}) = z_{ij}\ \Sigma_b\ z_{ik}^\top + (R_i)_{jk}.$$

The first term, $z_{ij}\ \Sigma_b\ z_{ik}^\top$ represents correlation induced by shared random effects, while the second term, $(R_i)_{jk}$ adds any residual serial correlation between observations j and k for subject i.

Within-subject correlation describes how repeated measures from the same individual covary over time. Between-subject correlation, by contrast, reflects the association between subject-level means, whether individuals with higher overall levels of one response tend to have higher levels of another, and is captured by the covariance of random intercepts in Σ_b.

Model structures

Let X_i and Z_i denote the design matrices obtained by stacking the row vectors $x_{ij}^\top$ and $z_{ij}^\top$, respectively, for all measurement times t_{ij} of subject i. Thus, z_{ij} is the j-th row of Z_i, and the term $Z_i \Sigma_b Z_i^\top$ represents the contribution of random effects to the marginal covariance of Y_i.

The overall (marginal) covariance of Y_i is

$$\Omega_i = Z_i \; \Sigma_b \; Z_i^\top + R_i,$$

which combines within-subject correlation from R_i and between-subject correlation from Σ_b.

Interpretation

Random intercepts capture persistent differences in overall levels across subjects, while random slopes capture differences in longitudinal trends. The residual covariance R_i accounts for remaining within-subject dependence not explained by these random effects, and the full marginal covariance Ω_i represents the combined temporal and cross-temporal dependence in the observed data. These components form the longitudinal submodel of the joint framework, which will be linked to the survival component through shared random effects in the next section.

Transformation model for survival data

The transformation survival model expresses the cumulative hazard through a monotonic transformation of the integrated risk, thereby unifying the proportional hazards and proportional odds frameworks within a single semiparametric formulation (Wang & Zhong, 2025; Wu et al., 2012).

Let $U_i(t) = x_i(t)^\top \beta + z_i(t)^\top b_i$ denote a latent process summarizing the evolving covariate trajectory (often introduced later as part of a longitudinal model), and let $W_i(t)$ be fully observed external covariates. The semiparametric transformation survival model specifies the cumulative hazard as

$$\Lambda_i(t \mid b_i) = H\left(A_i(t)\right), \quad A_i(t) = \int_0^t \lambda_0(s) \; \exp\{\eta_i(s)\} \; ds, \quad \eta_i(t) = \alpha \; U_i(t) + \phi^\top \; W_i(t),$$

where $H(\cdot)$ is a known, strictly increasing transformation, $\lambda_0(\cdot)$ is an unspecified baseline hazard, and α and ϕ are regression parameters capturing covariate effects on event risk.

The hazard follows by differentiation:

$$\lambda_i(t \mid b_i) = H^{'}(A_i(t)) \cdot \lambda_0(t) \cdot \exp\{\eta_i(t)\}.$$

The exponentiated term $\exp\{\eta_i(t)\}$ represents the relative risk at time t, while $H^{'}(\cdot)$ modulates how that risk evolves on the transformed cumulative hazard scale. The baseline hazard $\lambda_0(t)$ captures population-level event intensity and remains unspecified, preserving semiparametric flexibility.

The corresponding survival function is obtained from the relationship $S_i(t) = \exp\{-\Lambda_i(t)\}$, which expresses the probability of survival beyond time t as an exponential of the cumulative hazard.

Proportional hazards (PH)

If $H(u) = u$, then $H^{'}(u) = 1$ and

$$\lambda_i(t \mid b_i) = \lambda_0(t) \cdot \exp\{\eta_i(t)\}.$$

For time-fixed covariates, the hazard ratio between subjects is constant in t, satisfying the proportional hazards assumption. The PH model is preferred when the effect of covariates on the hazard appears approximately constant over time, common in biomedical and reliability studies with stable relative risks.

Proportional odds (PO)

If $H(u) = \log(1+u)$, then $H^{'}(u) = \frac{1}{1+u}$, giving

$$\Lambda_i(t) = \log\big(1 + A_i(t)\big), \quad S_i(t) = \exp\{-\Lambda_i(t)\} = \frac{1}{1 + A_i(t)}.$$

For time-fixed covariates, this yields proportional survival odds:

$$\frac{S_i(t)/(1 - S_i(t))}{S_{i'}(t)/(1 - S_{i'}(t))} = \exp\{-(\eta_i - \eta_{i'})\}.$$

Unlike the PH model, PO allows hazard ratios to vary over time while preserving proportionality on the survival-odds scale. It is preferred when the proportional hazards assumption fails but a monotonic relationship between covariates and survival is still plausible.

The choice between proportional hazards and proportional odds depends on whether covariate effects on risk appear constant over time or gradually diminish.

Box-Cox family

For $H_\rho(u) = \frac{\log(1+\rho u)}{\rho}$, we have $H_\rho^{'}(u) = \frac{1}{1+\rho u}$, and

$$\lambda_i(t \mid b_i) = \frac{\lambda_0(t)\ \exp\{\eta_i(t)\}}{1 + \rho A_i(t)}.$$

The limits $\rho \to 0$ and $\rho = 1$ recover PH and PO, respectively, with $0 < \rho < 1$ providing a continuum between the two. Intermediate values of ρ allow partial relaxation of the proportional hazards assumption while maintaining interpretability.

In practice, ρ is often selected by fitting the model across a grid of candidate values and comparing information criteria such as Akaike's Information Criterion (AIC) or Bayesian Information Criterion (BIC). The value minimizing AIC provides an empirical balance between proportional hazards and proportional odds structures.

Practical interpretation

The transformation model provides a flexible yet interpretable link between covariates and survival probabilities. Given estimated parameters $\hat{\beta}$, $\hat{\phi}$, $\hat{\alpha}$, and baseline hazard $\hat{\lambda}_0(t)$, the predicted survival probability for subject i at time t is

$$\widehat{S}_i(t) = \exp\big\{ - H(\widehat{A}_i(t))\big\}, \quad \widehat{A}_i(t) = \int_0^t \hat{\lambda}_0(s)\ \exp\{\hat{\eta}_i(s)\}\ ds.$$

This allows direct calculation of interpretable quantities such as:

- $P(T_i > t^*) = \hat{S}_i(t^*)$: probability of survival beyond a fixed time t^*,
- the median survival time, obtained by solving $\widehat{S}_i\left(t_{\frac{1}{2}}\right) = \frac{1}{2}$,
- contrasts between groups through ratios or differences of predicted survival or odds.

Under the PH specification, interpretation centers on constant relative risks via hazard ratios. Under the PO specification, interpretation shifts to survival-odds ratios and risk differences over time, often yielding more stable and clinically relevant measures when hazard functions cross.

Interpretation summary

Transformation models extend classical survival regression by:

- Retaining a nonparametric baseline hazard $\lambda_0(t)$,
- Allowing both proportional and non-proportional hazard relationships through $H(\cdot)$,
- Maintaining interpretable regression effects on hazard, odds, or probability scales.

They provide a coherent semiparametric framework for modeling time-to-event data and will serve as the survival submodel of the joint framework, in which $U_i(t)$, the latent longitudinal trajectory, is linked to the event process through shared random effects to form the full joint likelihood.

Joint model for longitudinal and survival data

The joint modeling framework combines the longitudinal mixed model and the transformation survival model through shared latent random effects $\mathbf{b}_i$, which capture the correlation between an individual's longitudinal trajectory and event risk (Wang & Zhong, 2025; Wu et al., 2012).

The joint model is:

$$\begin{cases} V_{ij} = U_i(T_{ij}) + \epsilon_{ij} = x_{ij}^\top \beta + z_{ij}^\top b_i + \varepsilon_{ij}, \\ \Lambda(t \mid \mathbf{b}_i, U_i, \mathbf{W}_i) = H_\rho \left(\int_0^t \lambda_0(s) \ \exp\{\alpha \ U_i(s) + \phi^\top \ \mathbf{W}_i(s)\} \ ds \right) \end{cases}$$

where V_{ij} denotes the observed longitudinal measurement at time t_{ij}; $U_i(t)$ is the underlying latent trajectory generated by the linear mixed model and indexed by the subject-specific random effects $\mathbf{b}_i$; $\mathbf{W}_i(t)$ represents fully observed external covariates; $\lambda_0(t)$ is the baseline hazard function; and $H_\rho(\cdot)$ is the transformation that determines the hazard scale, encompassing choices such as the proportional hazards, proportional odds, or Box–Cox specifications.

Shared random effects structure

The association between the two processes arises through $\mathbf{b}_i$, shared between the longitudinal and survival submodels. This induces dependence between the repeated-measure process and the event time by linking subject-specific deviations from the population trajectory to the hazard of event occurrence. The parameter α quantifies the strength and direction of this association:

- $\alpha > 0$ indicates that higher latent values $U_i(t)$ correspond to increased hazard,
- $\alpha < 0$ indicates a decreased hazard (protective effect).

The shared–parameter formulation can be implemented under two general estimation paradigms. In the two-stage approach, the longitudinal submodel is first fit separately to obtain empirical Bayes estimates of each subject's random effects $\hat{\mathbf{b}}_i$, which are then treated as known covariates in the survival model. This method is computationally simpler but can produce biased estimates because it ignores uncertainty in $\hat{\mathbf{b}}_i$ and the induced dependence between stages. The joint likelihood approach performs simultaneous estimation of all parameters by integrating over the random effects in a unified likelihood, ensuring valid inference and semiparametric efficiency under the assumed model (Wang & Zhong, 2025; Wu et al., 2012).

The following section details the likelihood formulation and estimation procedures for both the two-stage and joint approaches, including integration over random effects, treatment of censoring, and implementation via the EM algorithm.

Estimation and inference

Estimation proceeds under two complementary frameworks: a two-stage approach, which fits the longitudinal and survival components sequentially, and a full joint-likelihood approach, which estimates all parameters simultaneously by integrating over the latent random effects.

Two-stage approach

In the two-stage procedure (Wang & Zhong, 2025; Wu et al., 2012), the longitudinal and survival components are fit sequentially.

Stage 1: A linear mixed model is fitted to the repeated-measure data, producing empirical Bayes estimates of the subject-specific random effects $\hat{b}_i$ and corresponding predicted latent trajectories $U_i(t) = x_i(t)^\top \hat{\beta} + z_i(t)^\top \hat{b}_i$. These predicted current values are then carried forward as inputs to the survival analysis.

Stage 2: The survival submodel is fitted conditional on the $\hat{b}_i$, treating the predicted trajectory $U_i(t)$ as if it were observed without error. Because this ignores the sampling variability in $\hat{b}_i$, the resulting association estimate is typically attenuated and its standard errors underestimated. The problem is amplified when measurement error is large, longitudinal sampling is sparse, or dropout is strongly informative. The sequential fitting also breaks the likelihood coupling between submodels, rendering likelihood-based inference invalid.

Although computationally efficient and useful for exploratory analysis or initial parameter values, the two-stage approach does not provide valid inference for the association parameter. Accurate estimation requires simultaneous evaluation of the full joint likelihood.

Modified two-stage approach (bootstrap adjustment)

A modified two-stage approach can be used to correct for the loss of precision inherent in standard two-stage methods. After fitting the longitudinal and survival submodels separately, parameter uncertainty from the first stage is propagated to the second stage through bootstrap resampling (Wu et al., 2012).

In practice, this involves repeatedly resampling subjects with replacement from the original dataset. For each bootstrap sample, the longitudinal model is refit to obtain updated estimates of the random effects or predicted trajectories, which are then carried forward to the survival stage. The survival model is subsequently re-estimated using these resampled and refitted trajectories. This process is repeated several hundred times to capture sampling variability across both stages.

The bootstrap distributions of the estimated parameters are then used to compute empirical standard errors, confidence intervals, and bias corrections. This procedure allows the modified two-stage approach to account for uncertainty from Stage 1, producing more reliable inference while retaining computational efficiency relative to the full joint likelihood method.

Likelihood formulation

In general terms, the likelihood function represents the joint probability of observing the data, expressed as a function of the model parameters. It quantifies how well a given set of parameters explains the observed outcomes. In longitudinal–survival settings, the data consist of repeated longitudinal measurements and event-time outcomes that are typically correlated through subject-specific latent effects. The joint likelihood is therefore constructed by combining the conditional distributions of the longitudinal and survival components, linked through shared random effects b_i. Each subject contributes an individual likelihood obtained by multiplying the conditional densities of the longitudinal and survival responses given b_i and then integrating over the distribution of b_i to account for their unobserved nature. This marginalization is an application of the law of total probability: the observed-data likelihood is obtained by integrating the complete-data density $p_\psi(Y_i, \Delta_i \mid b_i)\; p_\theta(V_i \mid b_i)\; p_\theta(b_i)$ over the distribution of the latent random effects. This integration

yields the marginal likelihood of the observed data, forming the basis for maximum likelihood estimation of both fixed effects and variance components in the joint model.

$$L_n(\psi \mid O) = \prod_{i=1}^{n} \int p_\psi(Y_i, \Delta_i \mid b_i)\ p_\theta(V_i \mid b_i)\ p_\theta(b_i)\ db_i,$$

where $\psi = (\theta, \Lambda_0)$. Here, $\theta = (\alpha, \beta, \phi, \sigma_\epsilon^2, \Sigma_b)$ denotes the finite-dimensional parameter vector that governs the fixed effects and variance components of the longitudinal and survival submodels. Specifically, it includes the association parameter α, the fixed-effects coefficients β and ϕ, the residual variance σ_ϵ^2, and the random-effects covariance matrix Σ_b.

Λ_0 represents the baseline cumulative hazard function, an infinite-dimensional (nonparametric) component characterizing the underlying hazard structure of the survival process. It is related to the baseline hazard $\lambda_0(t)$ through $d\Lambda_0(t) = \lambda_0(t)\ dt$.

Together, $\psi = (\theta, \Lambda_0)$ collects both the parametric and nonparametric components of the joint model that are estimated through the nonparametric maximum likelihood approach.

$p_\psi(Y_i, \Delta_i \mid b_i)$ is the conditional likelihood contribution from the survival component of the joint model, given the subject-specific random effects b_i that link it to the longitudinal process. Y_i represents the observed or censored time, and Δ_i indicates whether an event occurred, $\Delta_i = 1$, or the observation was censored, $\Delta_i = 0$. This function specifies the likelihood of observing the event or censoring outcome for subject i conditional on their random effects.

$$p_\psi(Y_i, \Delta_i \mid b_i) = \Big[\lambda_0(Y_i) \exp\{\alpha\ U_i(Y_i) + \phi^\top\ W_i(Y_i)\}\ H^{'}(A_i(Y_i))\Big]^{\Delta_i} \cdot \exp\Big\{ - H(A_i(Y_i))\Big\},$$

$$A_i(Y_i) = \int_0^{Y_i} \lambda_0(s) \exp\{\alpha\ U_i(s) + \phi^\top\ W_i(s)\}\ ds.$$

$p_\theta(V_i \mid b_i)$ is the conditional longitudinal likelihood, representing the probability of subject i's observed longitudinal measurements given their random effects b_i. The term $V_{ij} - \beta^\top X_i(T_{ij}) - b_i^\top Z_i(T_{ij})$ is the residual in the LMM, corresponding to the deviation of the observed measurement from its model-predicted mean.

$$p_\theta(V_i \mid b_i) = \prod_{j=1}^{n_i} \frac{1}{(2\pi\sigma_\epsilon^2)^{1/2}} \exp\left\{ - \frac{\big(V_{ij} - \beta^\top X_i(T_{ij}) - b_i^\top Z_i(T_{ij})\big)^2}{2\sigma_\epsilon^2} \right\}.$$

$p_\theta(b_i)$ is the random-effects density, which defines how the subject-specific latent effects b_i are distributed in the population. Note that q is the dimension of the random-effects vector b_i (the number of random components), μ is the mean of the random effects, typically set to 0, and $|\cdot|$ denotes the determinant. This term represents the distributional assumption underlying the random effects, usually multivariate normal $b_i \sim N(\mu, \Sigma_b)$. It forms the mixing component of the joint likelihood, providing the population-level weighting for integrating over the unobserved b_i.

$$p_\theta(b_i) = \frac{1}{(2\pi)^{q/2}|\Sigma_b|^{1/2}}\ \exp\left\{-\tfrac{1}{2}(b_i - \mu)^\top \Sigma_b^{-1}(b_i - \mu)\right\}.$$

The joint likelihood links the longitudinal and survival processes through shared latent random effects, integrating over their distribution to obtain the marginal likelihood. This combines the Gaussian linear mixed

model and the transformation survival model within a semiparametric framework based on a nonparametric maximum likelihood estimate of the baseline hazard. Estimation is typically carried out using an expectation–maximization (EM) algorithm in which the E-step requires numerical approximation of the latent-effect integrals (e.g., adaptive quadrature or Laplace methods), and the M-step updates both regression parameters and the discrete baseline hazard. This approach yields semiparametric, efficient estimators for both the regression parameters and the nonparametric baseline hazard.

EM algorithm

Maximum likelihood estimation proceeds via an expectation–maximization (EM) algorithm, treating the subject-specific random effects b_i as latent variables (Zhai, 2007; Chen & Gupta, 2010). Let $\psi = (\theta, \Lambda_0)$ denote the full parameter vector, and let $O_i = (V_i,\, Y_i,\, \Delta_i)$ denote the observed data for subject i, consisting of the longitudinal measurements $V_i = \{V_{ij}\}$, the observed event or censoring time $Y_i = \min(T_i, C_i)$, and the event indicator $\Delta_i = I(T_i \le C_i)$. The complete-data log-likelihood is

$$\ell_c(\psi) = \sum_{i=1}^{n} \Big\{ \log p_\psi(Y_i, \Delta_i \mid b_i) + \log p_\theta(V_i \mid b_i) + \log p_\theta(b_i) \Big\},$$

Given current values $\psi^{(k)}$, the EM algorithm alternates between the following steps.

E-step

Compute the conditional expectation of the complete-data log-likelihood,

$$Q(\psi \mid \psi^{(k)}) = \sum_{i=1}^{n} E_{b_i \mid O_i, \psi^{(k)}}[\ell_c(\psi)],$$

where the expectations with respect to $p(b_i \mid O_i, \psi^{(k)})$ are evaluated numerically (e.g., adaptive Gauss–Hermite quadrature or Laplace approximation). In practice, this reduces to computing low-dimensional moments such as

$$E(b_i \mid O_i, \psi^{(k)}) \quad \text{and} \quad E(b_i b_i^\top \mid O_i, \psi^{(k)}),$$

along with the corresponding expectations of the survival risk terms.

M-step

Maximize $Q(\psi \mid \psi^{(k)})$ with respect to ψ. The updates separate naturally:

- Finite-dimensional parameters θ:

Updated using the E-step moments, typically through weighted score equations or a Newton–Raphson step. For the longitudinal model this corresponds to generalized least squares with the conditional expectations of b_i.

- Baseline cumulative hazard Λ_0:

Updated nonparametrically with discrete jumps at the event times. The EM updates take a Breslow-type form, with each jump equal to the ratio of observed events to the posterior-averaged risk at that time, yielding the NPMLE of Λ_0.

The E- and M-steps iterate until convergence of $\psi^{(k)}$. Under correct specification, the resulting estimators

of the finite-dimensional parameters achieve semiparametric efficiency. Because the model includes a nonparametric baseline hazard, bootstrap methods are typically preferred for assessing variability.

Simulation study

This simulation study evaluates how accurately each estimator recovers the association parameter α. We compare the bias and efficiency of the two-stage and joint-likelihood approaches, assess robustness across varying measurement error, association strength, and censoring levels, and examine how the transformation choice H_ρ influences estimation.

We begin by locking in a single joint model data generating process (DGP) as a baseline framework.

Data generating process for joint model

The following describes the mechanism for generating one synthetic dataset. In all simulation scenarios, this procedure is repeated independently for R Monte Carlo replicates, with new random effects, longitudinal trajectories, measurement errors, event times, and censoring times drawn at each replicate.

Observation grid

We generate repeated measurements on a fixed observation grid. For each subject $i = 1, \ldots, n$, longitudinal responses are recorded at prespecified times t_{ij} drawn from a common grid, for example $t_{ij} \in \{0, 1, 2, \ldots, 9\}$ ($t_{max} = 9$). This ensures identical measurement occasions across subjects and isolates longitudinal–survival dependence from complications due to irregular sampling. Unless otherwise noted, simulations use $n = 400$.

Random effects

For each subject, the random intercept and random slope are generated as

$$b_i = \begin{bmatrix} b_{0i} \\ b_{1i} \end{bmatrix} = \begin{bmatrix} \sigma_{b_0} z_1 \\ \rho_b \sigma_{b_1} z_1 + \sqrt{1 - \rho_b^2} \sigma_{b_1} z_2 \end{bmatrix} \sim N(0, \Sigma_b)$$

where $z_1, z_2 \overset{\text{iid}}{\sim} N(0, 1)$. This construction yields the covariance matrix

$$\Sigma_b = \begin{pmatrix} \sigma_{b0}^2 & \rho_b \sigma_{b0} \sigma_{b1} \\ \rho_b \sigma_{b0} \sigma_{b1} & \sigma_{b1}^2 \end{pmatrix}.$$

For the baseline simulation design, we set

$$\sigma_{b0}^2 = 1, \quad \sigma_{b1}^2 = 0.3^2, \quad \rho_b = 0.3,$$

yielding moderate intercept variability, smaller slope variability, and a positive intercept–slope correlation.

Latent longitudinal process and measurement model

The latent trajectory is generated from

$$U_i(t) = \beta_0 + \beta_1 t + b_{0i} + b_{1i} t,$$

with baseline parameter choices $\beta_0 = 0$ and $\beta_1 = -0.1$.

The observed longitudinal measurements are then generated from the measurement model

$$V_{ij} = U_i(t_{ij}) + \varepsilon_{ij}, \quad \varepsilon_{ij} \sim N(0, \sigma_\varepsilon^2),$$

with baseline noise level $\sigma_\varepsilon = 0.3$. Later scenarios vary σ_ε to assess robustness under increasing measurement error.

Survival model

Event times are generated under the proportional hazards specification with a Weibull baseline hazard. This serves as the baseline transformation-model scenario; later simulations vary the association α, measurement-error variance σ_ε^2, the censoring distribution, and the transformation H_ρ to assess robustness and misspecification. A Weibull baseline is used because it provides a flexible parametric form capable of representing increasing or decreasing hazards while remaining analytically tractable, computationally stable, and fully compatible with the transformation-model framework.

The subject-specific hazard is

$$\lambda_i(t \mid U_i) = \lambda_0(t) \exp\{\alpha U_i(t)\},$$

with a Weibull baseline

$$\lambda_0(t) = \kappa \lambda t^{\kappa - 1}, \quad \kappa = 1.3.$$

The scale parameter λ is chosen so that a baseline trajectory $U_i(t) = 0$ yields

$$\exp\{-\lambda t_{\max}^{\kappa}\} = 0.5 \quad \text{at} \quad t_{max} = 9,$$

producing an approximate 50% event rate in the absence of censoring. This intermediate regime ensures enough events to estimate the association while avoiding nearly deterministic early failure. To isolate the contribution of the internal covariate, no external covariates are included in the baseline design, $W_i(t) = 0$.

The cumulative hazard and survival function follow directly as

$$A_i(t) = \int_0^t \lambda_0(s) \exp\{\alpha U_i(s)\}\, ds, \quad \Lambda_i(t) = A_i(t), \quad S_i(t) = \exp\{-\Lambda_i(t)\}.$$

Inverse transform sampling

A convenient way to generate event times under the specified proportional hazards model is to use inverse transform sampling. For any survival function $S_i(t)$, the defining property is that the event time T_i satisfies $P(T_i > t) = S_i(t)$. If $W_i \sim \text{Unif}(0, 1)$, then the random variable that solves $S_i(T_i) = W_i$ has survival function S_i. Under the proportional hazards specification, $S_i(t) = \exp\{-A_i(t)\}$, so this condition becomes

$$\exp\{-A_i(T_i)\} = W_i \quad \Longleftrightarrow \quad A_i(T_i) = -\log W_i.$$

Thus, for each subject we draw one uniform random variable W_i and determine T_i as the unique time at which the cumulative hazard reaches the value $-\log W_i$. Because $A_i(t)$ depends on the subject-specific latent trajectory $U_i(s)$, the resulting equation has no closed-form solution and must be evaluated numerically.

In practice, computing T_i reduces to solving a simple one-dimensional nonlinear equation. After drawing $W_i \sim \text{Unif}(0,1)$, we set $c_i = -\log W_i$, and define $f_i(t) = A_i(t) - c_i$. Because $A_i(t)$ is continuous and nondecreasing, the equation $f_i(t) = 0$ has at most one solution. Numerically, we search for this root on a large finite interval $[0, t_{\text{cap}}]$. If $A_i(t_{\text{cap}}) \geq c_i$, a unique solution exists in $(0, t_{\text{cap}})$ and is found using a bracketing method (e.g., uniroot in R). If $A_i(t_{\text{cap}}) < c_i$, the subject's hazard remains so low that the event effectively occurs beyond the study horizon, and we set $T_i = t_{\text{cap}}$.

Censoring and truncation

To mimic incomplete follow-up, we generate independent censoring times C_i from an exponential distribution with rate chosen to yield a desired censoring proportion (e.g., 5%, 10%, or 25%). The observed time and event indicator are $Y_i = \min(T_i, C_i, t_{max})$ and $\Delta_i = I\{T_i \leq C_i \cap T_i \leq t_{\max}\}$. Longitudinal measurements taken after Y_i are discarded, i.e., measurements with $t_{ij} > Y_i$ are removed. This induces event-driven dropout of the longitudinal process and reproduces the informative truncation handled by the joint model. Delayed entry (left truncation) was not considered in this simulation.

Methods

Each simulated dataset is analyzed under two estimation strategies applied to the same joint model specified in the preceding sections: a full joint-likelihood estimator, treated as the reference ("gold-standard"), and a conventional two-stage estimator. The primary estimand is the association parameter α linking the latent longitudinal trajectory $U_i(t)$ to the event hazard. To monitor model performance more broadly, we also summarize the fixed-effects parameters (β_0, β_1), the measurement-error variance σ_ε^2, and the random-effects covariance components.

Joint-likelihood estimator (gold standard)

The joint-likelihood estimator fits the longitudinal and survival submodels simultaneously in a shared-random-effects framework. The longitudinal component is a linear mixed-effects model with random intercept and slope on time, fitted to the longitudinal measurements truncated at the observed time Y_i. The survival component is a proportional hazards model with a current-value association, where the hazard at time t depends on the latent trajectory $U_i(t)$ through $\exp\{\alpha U_i(t)\}$.

Joint estimation proceeds by integrating over the latent random effects and maximizing the joint likelihood, as described previously in the likelihood and EM sections. The resulting estimator of the association parameter, $\widehat{\alpha}_{\text{joint}}$, is taken as the posterior mean of the current-value coefficient in the fitted joint model. All simulation replicates produced valid fits for the mixed-effects and Cox components, and the joint model converged in each dataset.

Two-stage estimator (plug-in internal covariate)

The two-stage estimator applies the same sequential fitting strategy used in the general two-stage procedure. In Stage 1, the linear mixed model is fitted to the truncated longitudinal data to generate empirical Bayes predictions of the latent trajectories, $\widehat{U}_i(t)$. In Stage 2, these predicted trajectories are incorporated as a time-dependent internal covariate in a proportional hazards model written in counting-process form. The coefficient on $\widehat{U}_i(t)$ yields the estimate $\widehat{\alpha}_{\text{two}}$.

Because $\widehat{U}_i(t)$ is treated as error-free, the estimator typically misstates the true association, with attenuation increasing under higher measurement error, sparser longitudinal sampling, or more informative dropout. In the simulation it serves as a benchmark against which the joint estimator's correction of this attenuation can be evaluated.

All simulation replicates produced valid fits for both stages, and no datasets were discarded.

Performance evaluation

For each simulated dataset and each method, we record:

- $\widehat{\alpha}_{\text{joint}}$ and $\widehat{\alpha}_{\text{two}}$,
- the estimated fixed-effects parameters $(\widehat{\beta}_0, \widehat{\beta}_1)$,
- the estimated residual standard deviation $\widehat{\sigma}_\varepsilon$,
- the estimated random-effects covariance components.

Although α is the primary estimand, examining the longitudinal parameters ensures that both estimation strategies consistently recover the underlying mixed-effects structure across simulation settings. Because every replication yielded valid parameter estimates, all datasets contribute to the Monte Carlo summaries.

For each scenario, we report empirical means, standard deviations, and biases of the estimators, providing a comprehensive assessment of accuracy, efficiency, and robustness across varying levels of measurement error, association strength, and censoring.

Monte Carlo implementation

Each scenario was simulated with R=30 Monte Carlo replications. At this replication level, Monte Carlo standard errors are small relative to the biases of interest $MCSE(\bar{\theta}) = \text{SD}/\sqrt{R}$ is typically about 0.01 for the parameters considered. This size is sufficient for identifying qualitative patterns in bias and variability across the simulation grid while keeping computation feasible; the joint estimator requires costly likelihood evaluations, so substantially larger R would offer limited additional insight for the present study. In a publication setting, we would increase the replication size (e.g., to $R \approx 1000$) to render MCSE negligible.

For each combination of simulation parameters $(\sigma_\varepsilon, \alpha, \text{censoring rate}, \rho)$, we generated $R = 30$ independent datasets. For every dataset:

- The longitudinal mixed model was fit to the truncated repeated-measure data.
- The two-stage estimator $\widehat{\alpha}_{\text{two}}$ was obtained by fitting a Cox model with the predicted current value $\widehat{U}_i(t)$ included as a time-dependent internal covariate.
- The joint estimator $\widehat{\alpha}_{\text{joint}}$ was obtained by fitting the shared random-effects joint model.

All fits converged successfully, with no estimation failures or discarded datasets.

Software

All simulations and analyses were conducted in R version 4.5.0 (2025-04-11) using RStudio version 2025.9.1.401 ("Cucumberleaf Sunflower"). Estimation relied on the *tidyverse* ecosystem for data manipulation and graphics, *nlme* for linear mixed-effects models, *survival* for proportional hazards models, and *JMbayes2* for shared-parameter joint models. Parallel computation was implemented through the *future* and *furrr* frameworks using a multisession backend to distribute Monte Carlo replications across available CPU cores. Full citations for R, RStudio, and all packages used appear in the Bibliography.

Results

This simulation study evaluates the ability of each estimator to recover the association parameter α, with emphasis on bias, variability, and robustness across measurement error levels, association strengths, and censoring rates. Two estimation strategies are compared: the joint-likelihood estimator ("joint") and the conventional two-stage estimator ("two-stage"). Unless otherwise noted, results are reported under the correctly specified proportional hazards (PH) model with transformation parameter $\rho = 0$.

Baseline scenario

The baseline configuration fixes the association at $\alpha = 0.5$ with low measurement error ($\sigma_\varepsilon = 0.3$), 10% independent censoring, and a correctly specified proportional hazards structure ($\rho = 0$). Table 1 reports estimator performance under this setting. The joint estimator is effectively unbiased, with mean estimates centered tightly around the true value and only modest Monte Carlo variability, yielding low overall error. In contrast, the two-stage estimator systematically overestimates the association parameter. Although its sampling variability is comparable to that of the joint estimator, the persistent upward bias leads to a slightly larger RMSE. This pattern establishes the joint estimator as the accuracy benchmark for subsequent comparisons and highlights the inherent attenuation and scale distortion affecting the two-stage approach even under mild measurement error and minimal censoring.

Table 1: Baseline scenario ($\alpha = 0.5$, $\sigma_\varepsilon = 0.3$, censoring = 10%, $\rho = 0$): Comparison of joint and two-stage estimators.

Estimator	True α	Mean	SD	Bias	RMSE
Joint likelihood	0.5	0.5095	0.0494	0.0095	0.0495
Two-stage	0.5	0.5347	0.0522	0.0347	0.0619

Measurement error

Table 2 summarizes estimator performance across the three measurement-error levels in the simulation grid. Under the correctly specified PH model with 10% censoring, the joint estimator remains stable across all values of σ_ε. Its mean estimates remain close to the true association and exhibit a uniformly slight upward bias, while both bias and RMSE show only mild sensitivity to increasing measurement noise. Although variance rises modestly at the moderate noise level, overall error remains essentially unchanged across the moderate and high measurement-error settings.

The two-stage estimator exhibits a distinctly monotone pattern across the measurement-error grid. As noise increases, its estimated association decreases smoothly: it begins with upward bias at the lowest noise level, remains slightly upward-biased but close to the truth at intermediate noise, and becomes downward-biased once the measurement error is large. This progression reflects a negative monotone drift driven by increasing misalignment between the predicted trajectories and the true latent process. Although the intermediate noise level yields a brief region in which variance and RMSE improve relative to the joint estimator, the overall behavior remains non-robust. This trajectory underscores the intrinsic sensitivity of the two-stage procedure to measurement error and the instability that emerges as σ_ε grows.

Overall, the joint estimator remains robust across the entire measurement-error grid, whereas the two-stage estimator exhibits non-robust behavior, with substantially greater fluctuation and instability around the true association.

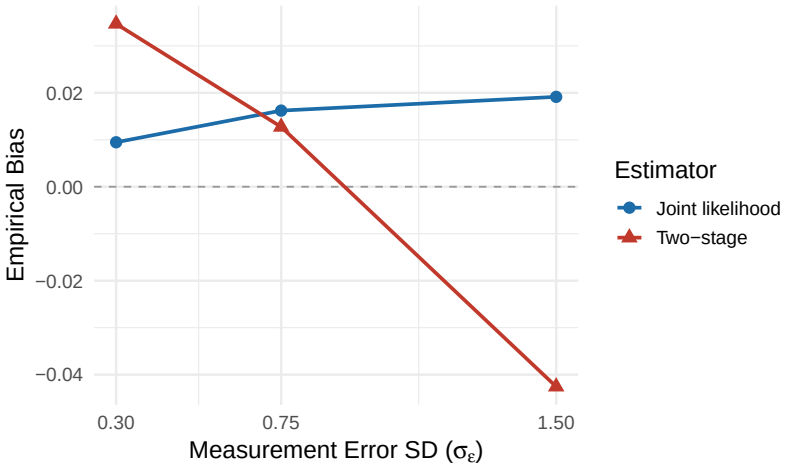
Table 2: Effect of measurement error on estimation of $\alpha = 0.5$ under the PH model ($\rho = 0$) with 10% censoring.

σ_ε	Estimator	True α	Mean	SD	Bias	RMSE
0.30	Joint likelihood	0.5	0.5095	0.0494	0.0095	0.0495
0.30	Two-stage	0.5	0.5347	0.0522	0.0347	0.0619
0.75	Joint likelihood	0.5	0.5162	0.0726	0.0162	0.0732
0.75	Two-stage	0.5	0.5127	0.0719	0.0127	0.0718
1.50	Joint likelihood	0.5	0.5191	0.0707	0.0191	0.0721
1.50	Two-stage	0.5	0.4574	0.0549	-0.0426	0.0688

To further illustrate these patterns, Figure 1 displays empirical bias for both estimators across the three measurement-error levels. The joint estimator shows consistently small upward bias with minimal sensitivity to increasing noise, whereas the two-stage estimator traces the negative monotone drift described above, beginning with upward bias, moving close to unbiased at moderate noise, and ultimately becoming downward-biased at the highest noise level. The figure reinforces the contrasting stability profiles of the two estimators and visually highlights the structural sensitivity of the two-stage approach to measurement error.

Figure 1

Effect of Measurement Error on Bias of α Estimates



Association strength

Table 3 summarizes estimator performance across the association strengths considered in the simulation grid, holding the PH model correctly specified ($\rho = 0$), measurement error low ($\sigma_\varepsilon = 0.3$), and censoring at 10%. Across this range of α, the joint estimator remains well behaved: its mean estimates track the true association closely, exhibiting a uniformly slight upward bias and only modest increases in variability as the association strengthens. For the two smaller association levels, ($\alpha = 0.25$ and $\alpha = 0.5$, the sampling variability is low for both estimators, and SD and RMSE are nearly indistinguishable; differences at these settings largely reflect Monte Carlo noise. A larger increase in error appears only at the strongest association level, where the contribution of the survival process to the joint likelihood is more pronounced, making the estimators more sensitive to discrepancies between the observed longitudinal paths and the underlying latent process.

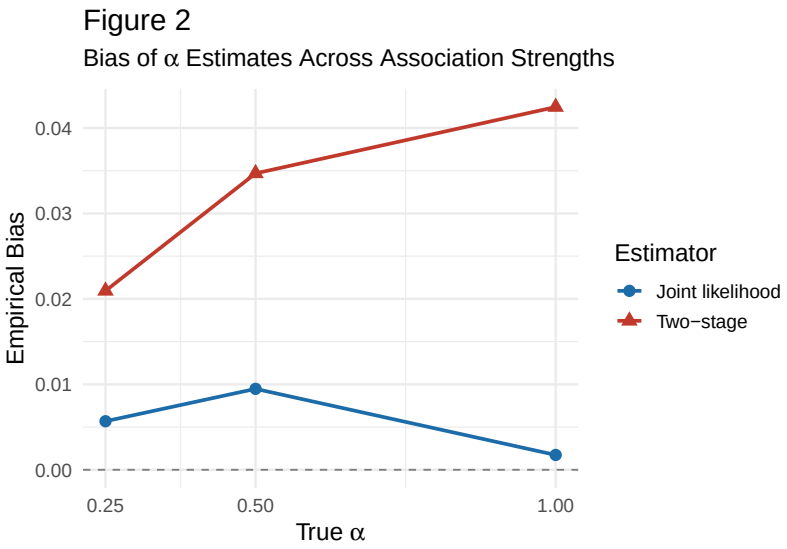
The two-stage estimator displays a systematically distorted response to increasing association strength. Its bias is modest at weaker associations but persists across the grid and becomes more pronounced as α grows, reflecting the same attenuation and scale distortion observed in the baseline scenario. Because this bias does not vanish, the two-stage estimator's RMSE becomes increasingly dominated by its systematic component rather than sampling variability, and the separation between the two estimators becomes clearer at higher association levels. Overall, while both methods show minimal differences in variability at low to moderate association strengths, the joint estimator provides more reliable calibration across the full grid, whereas the two-stage estimator becomes increasingly misaligned as the true association intensifies.

To visualize these patterns, Figure 2 plots empirical bias as a function of the true association strength. The joint estimator shows consistently small, slightly positive bias across the grid, whereas the two-stage estimator exhibits a persistent and more pronounced deviation from zero that grows with α. This figure reinforces the table-based comparison, highlighting that the two-stage procedure becomes increasingly miscalibrated as the underlying association strengthens.

Table 3: Effect of association strength on estimation of α under the PH model ($\rho = 0$), with $\sigma_\varepsilon = 0.3$ and 10% censoring.

True α	Estimator	Mean	SD	Bias	RMSE
0.25	Joint likelihood	0.2557	0.0547	0.0057	0.0541
0.25	Two-stage	0.2709	0.0576	0.0209	0.0603
0.50	Joint likelihood	0.5095	0.0494	0.0095	0.0495
0.50	Two-stage	0.5347	0.0522	0.0347	0.0619
1.00	Joint likelihood	1.0017	0.0679	0.0017	0.0668
1.00	Two-stage	1.0425	0.0702	0.0425	0.0810

Figure 2



Censoring

Table 4 summarizes estimator performance across the censoring levels in the simulation grid for the baseline association $\alpha = 0.5$, holding the PH model correctly specified ($\rho = 0$) and measurement error at its baseline value ($\sigma_\varepsilon = 0.3$). Across censoring rates, the joint estimator remains well behaved: its mean estimates stay close to the true association, with small, slightly positive bias and a gradual increase in variability as censoring intensifies. As expected, heavier censoring reduces effective information and inflates variance and RMSE, but this deterioration is modest at $\alpha = 0.5$, indicating that the joint procedure is relatively robust to loss of follow-up in this baseline setting.

The two-stage estimator shows a more pronounced sensitivity to censoring. Its bias, already present in the lightly censored setting, becomes more marked at higher censoring levels. Together with the loss of information induced by censoring, this persistent bias leads to a sharper rise in RMSE for the two-stage estimator compared with the joint estimator. Thus, while both methods suffer some loss of efficiency under heavier censoring, the joint estimator remains comparatively stable, whereas the two-stage estimator becomes less reliable as the proportion of censored observations grows.

To visualize how these patterns extend across association strengths, Figure 3 plots empirical bias as a function of the censoring rate for each value of α. The joint estimator exhibits consistently small bias across censoring levels, with only mild drift as censoring increases, whereas the two-stage estimator shows a persistent deviation from zero that can worsen at higher censoring rates. Notably, within each value of the true α, the joint and two-stage curves follow almost parallel trajectories across censoring rates; censoring shifts

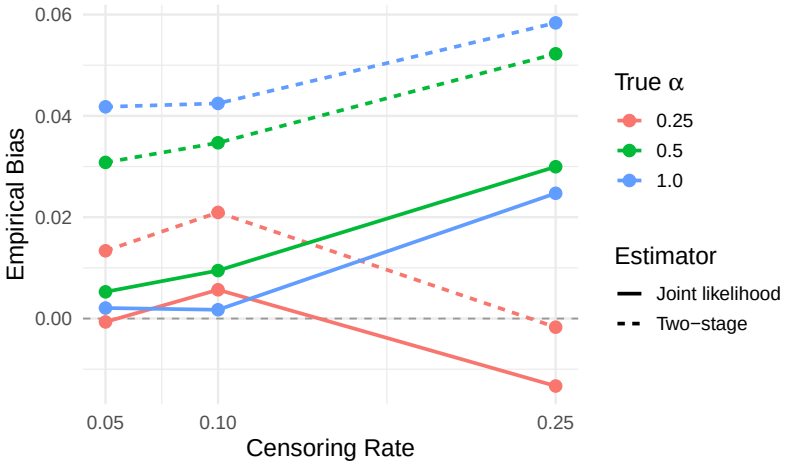
Table 4: Effect of censoring on estimation of the association parameter α under the PH model ($\rho = 0$) with baseline measurement error $\sigma_\varepsilon = 0.3$.

True α	Censoring	Estimator	Mean	SD	Bias	RMSE
0.5	0.05	Joint likelihood	0.5053	0.0370	0.0053	0.0367
0.5	0.05	Two-stage	0.5308	0.0388	0.0308	0.0490
0.5	0.10	Joint likelihood	0.5095	0.0494	0.0095	0.0495
0.5	0.10	Two-stage	0.5347	0.0522	0.0347	0.0619
0.5	0.25	Joint likelihood	0.5300	0.0751	0.0300	0.0796
0.5	0.25	Two-stage	0.5523	0.0757	0.0523	0.0909

the level of bias but leaves the overall trend largely unchanged. This reinforces the table-based comparison: censoring amplifies the two-stage procedure's bias while the joint estimator remains comparatively robust.

Figure 3

Bias of α Across Censoring Levels and Association Strengths



Transformation misspecification

Table 5 summarizes the impact of transformation misspecification on estimation of the association parameter $\alpha = 0.5$ under mild measurement error ($\sigma_\varepsilon = 0.3$) and 10% censoring, comparing the correctly specified proportional hazards model ($\rho = 0$) to the proportional odds specification ($\rho = 1$). Under the correct PH transformation, both estimators show a clear bias that increases with the strength of the association. When the PO transformation is imposed, the magnitude of the bias grows substantially, shifting from mild positive bias under PH to pronounced negative bias, and this deterioration becomes more severe as α increases.

The two-stage estimator is more biased than the joint estimator when the PH model is correctly specified, whereas under the misspecified PO transformation the joint estimator becomes the more biased of the two. Overall, both procedures are sensitive to the transformation choice, but the joint approach maintains better performance under correct specification and degrades sharply under misspecification, while the two-stage estimator performs poorly in both settings.

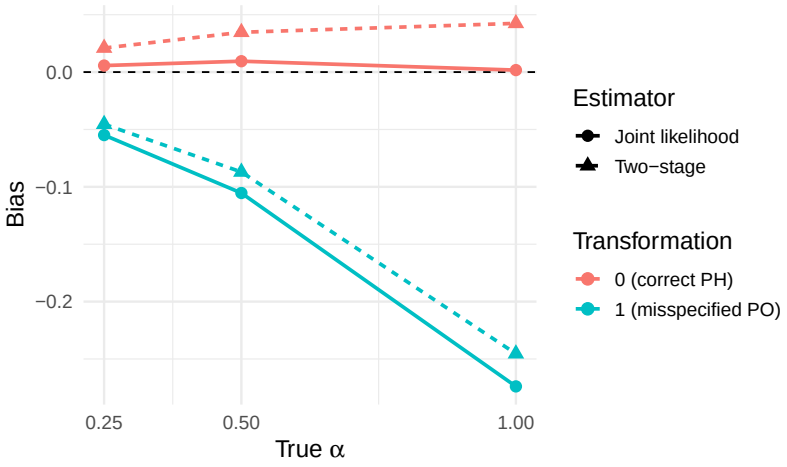
Figure 4 displays these patterns graphically, plotting empirical bias against the true α for $\rho = 0$ and $\rho = 1$ under 10% censoring and mild measurement error ($\sigma_\varepsilon = 0.3$). Both estimators are materially biased, and the bias becomes more pronounced as the association strength increases. Under the correctly specified PH transformation ($\rho = 0$), both joint and two-stage estimators exhibit small but positive bias with the

Table 5: Effect of transformation misspecification on estimation of the association parameter α with $\sigma_\varepsilon = 0.3$ and 10% censoring: comparison of PH ($\rho = 0$) and PO ($\rho = 1$) specifications.

True α	True ρ	ρ	Estimator	Mean	SD	Bias	RMSE
0.5	0	0	Joint likelihood	0.5095	0.0494	0.0095	0.0495
0.5	0	0	Two-stage	0.5347	0.0522	0.0347	0.0619
0.5	0	1	Joint likelihood	0.3946	0.0539	-0.1054	0.1179
0.5	0	1	Two-stage	0.4129	0.0562	-0.0871	0.1031

joint estimator very closely estimating the true α. Under the misspecified PO transformation ($\rho = 1$), the bias becomes strongly negative and increases in magnitude with α, with the joint estimator consistently lying farther from zero than the two-stage estimator. The parallel trajectories indicate that both estimators deteriorate in a similar way as α increases; misspecification shifts the bias substantially but does not change the overall trend.

Figure 4



Longitudinal model recovery

Table 6 summarizes recovery of the longitudinal model parameters $(\beta_0, \beta_1, \sigma_\varepsilon)$ under the correctly specified PH transformation ($\rho = 0$) with association $\alpha = 0.5$ and 10% censoring, across the three measurement–error settings $\sigma_\varepsilon \in \{0.30, 0.75, 1.50\}$. For each value of σ_ε the Monte Carlo mean, standard deviation, and empirical bias are reported for all three parameters.

Across measurement–error levels, the regression coefficients β_0 and β_1 display modest but noticeable bias, with bias magnitude increasing slightly as σ_ε grows. In contrast, estimation of the measurement–error standard deviation σ_ε is less sensitive: its empirical bias remains very small across all settings, and its Monte Carlo variability is comparatively low relative to the regression coefficients. Overall, the longitudinal component is well recovered, with σ_ε estimated particularly accurately even as measurement error increases.

Random-effects recovery

Table 7 reports recovery of the random-effects covariance components under the baseline scenario ($\alpha = 0.5$, $\sigma_\varepsilon = 0.3$, 10% censoring, $\rho = 0$). The marginal variances of the random intercept and slope, $\mathrm{var}(b_0)$ and $\mathrm{var}(b_1)$, and their covariance $\mathrm{cov}(b_0, b_1)$ are all estimated with minimal bias: the Monte Carlo means lie very close to the true values, and the empirical standard deviations are modest.

Table 6: Recovery of longitudinal parameters (β_0, β_1, and σ_ε) under the PH model ($\rho = 0$), with $\alpha = 0.5$ and 10% censoring.

σ_ε	Parameter	True Value	Mean	SD	Bias
0.3	β_0	0.0	0.0058	0.0498	0.0058
0.3	β_1	-0.1	-0.1061	0.0190	-0.0061
0.3	σ_ε	0.3	0.3008	0.0055	0.0008

Overall, these results indicate that, under the baseline configuration, the joint model recovers the random-effects covariance structure accurately, suggesting that the dependence between individual trajectories is well captured by the fitted model.

Table 7: Recovery of random-effects covariance components under the baseline scenario ($\alpha = 0.5$, $\sigma_\varepsilon = 0.3$, censoring = 10%, $\rho = 0$).

Parameter	True Value	Mean	SD	Bias
$\text{var}(b_0)$	1.00	1.0081	0.0720	0.0081
$\text{var}(b_1)$	0.09	0.0891	0.0070	-0.0009
$\text{cov}(b_0, b_1)$	0.09	0.0892	0.0175	-0.0008

Summary

Across all simulation scenarios, the joint likelihood estimator consistently provides accurate and stable recovery of the association parameter, with small, slightly positive bias and modest sensitivity to measurement error, association strength, censoring, or model configuration. In contrast, the two-stage estimator displays systematic distortion: it is upward-biased under mild error, drifts negatively as measurement noise increases, and exhibits increasing miscalibration as the true association strengthens or censoring grows. Both estimators are highly sensitive to transformation misspecification, but the joint approach performs well under the correct PH model and degrades sharply only when the transformation is wrong, whereas the two-stage estimator performs poorly in both settings. Recovery of the longitudinal trajectory parameters and the random-effects covariance structure is strong throughout, with only mild increases in bias under heavier measurement error. Taken together, the results highlight the robustness and reliability of the joint estimator and the structural instability of the two-stage procedure across realistic data-generating conditions.

Discussion

This simulation study demonstrates clear and systematic differences between the joint likelihood estimator and the conventional two-stage approach for recovering the longitudinal–survival association. Across a broad grid of measurement error, association strength, and censoring levels, the joint estimator shows strong and consistent performance: it remains nearly unbiased under correct model specification, its variability grows predictably with information loss, and its overall error profile remains stable even in settings with substantial noise or moderate censoring. These properties reflect its correct treatment of the latent trajectory and its coherent use of both longitudinal and survival information.

In contrast, the two-stage estimator exhibits structural limitations that persist throughout the simulation grid. Its bias is present even under favorable conditions and becomes increasingly pronounced under heavier measurement error, stronger associations, and higher censoring levels. The monotone bias drift observed across several scenarios indicates that the two-stage procedure does not simply lose efficiency but instead introduces systematic distortion in the estimated association. This behavior aligns with theoretical concerns about the imperfect alignment between predicted and true underlying trajectories and the propagation of

measurement error into the survival model when the predicted latent process is treated as error-free. These issues persist across all scenarios examined here, indicating that the distortions arise from the structure of the two-stage procedure rather than from Monte Carlo noise.

The study also highlights the importance of correct transformation specification. While both estimators perform well under the proportional hazards model, misspecifying the transformation introduces large, directional bias for both procedures. Notably, the joint estimator, strong under correct specification, becomes markedly biased under PO misspecification, while the two-stage estimator remains biased in both settings. This reinforces that reliable estimation of the longitudinal–survival association depends critically on the correct functional form of the hazard, and even minor departures from the true transformation can have substantial downstream effects.

Despite these differences in association recovery, both estimators demonstrate good performance in reconstructing the longitudinal process itself. The fixed effects and measurement-error variance are recovered with modest bias, and the random-effects covariance structure is estimated accurately under baseline conditions. This suggests that the challenges faced by the two-stage estimator do not arise from instability in the longitudinal model but from the disconnect between predicted trajectories and the true latent process when those predictions are later incorporated into the survival component.

Overall, the findings underscore the practical advantages of joint modeling when accurate estimation of the longitudinal–survival association is required. The joint estimator offers substantially greater robustness to measurement error, censoring, and changes in the association strength, and it deteriorates only under clear misspecification of the hazard transformation. The two-stage estimator, while simple to implement, consistently exhibits persistent and non-negligible bias across realistic data-generating conditions. For applications in which the association parameter is central to inference, such as biomarker validation, dynamic risk prediction, or studies of disease progression, the joint likelihood approach provides a far more reliable foundation.

Conclusion

This project developed and evaluated likelihood-based estimators for a transformation joint model linking a longitudinal internal covariate to a time-to-event outcome through shared random effects. Using a simulation design grounded in a realistically parameterized mixed model and proportional hazards survival process, we compared a full joint-likelihood estimator to a conventional two-stage procedure across varying levels of measurement error, association strength, censoring, and transformation specification.

Across all scenarios under the correctly specified proportional hazards model, the joint likelihood estimator consistently recovered the association parameter with small bias and stable variability, while simultaneously providing accurate estimates of the longitudinal fixed effects, measurement-error variance, and random-effects covariance structure. In contrast, the two-stage estimator exhibited persistent and sometimes substantial bias in the association parameter, with a characteristic monotone drift as measurement error increased and as the true association strengthened. These distortions did not reflect instability of the longitudinal fits but rather the structural limitations of treating predicted latent trajectories as error-free covariates in the survival model.

The transformation experiments further underscored the importance of correctly specifying the hazard scale. Both estimators were sensitive to misspecification of the transformation H_ρ, with the joint estimator performing very well under the correct proportional hazards model and deteriorating sharply when a proportional odds transformation was incorrectly imposed. This sensitivity highlights a key practical message: even when a joint model is used, accurate inference on the longitudinal–survival association depends critically on choosing an appropriate hazard structure.

Several limitations qualify these findings. The simulation was based on a single underlying DGP with linear trajectories, a Weibull baseline hazard, moderate random-effects complexity, and independent censoring;

more complex designs (e.g., nonlinear longitudinal trajectories, time-varying external covariates, or informative censoring) may reveal additional subtleties. The number of Monte Carlo replications was chosen to balance numerical precision with computational cost and would need to be increased in a publication-scale study. Finally, we considered only one form of two-stage estimator; bootstrap-adjusted or more elaborate plug-in methods may partially mitigate, but not fully eliminate, the structural biases identified here.

Taken together, the results provide clear guidance for practice. When the scientific goal is to quantify the association between an internal longitudinal covariate and event risk, full joint-likelihood estimation within an appropriately specified transformation model is strongly preferred. Two-stage approaches, while convenient, can introduce substantial and directionally predictable bias even in relatively benign settings. For applications such as biomarker validation and dynamic risk prediction, where accurate calibration of the longitudinal, survival link is central, joint modeling on a well-chosen hazard scale offers a far more reliable foundation for inference and decision-making.

Bibliography

1. Gut, Allan. An Intermediate Course in Probability. 2nd ed., Springer, 2009.

2. Wang, J.-L. and Zhong, Q. "Joint Modeling of Longitudinal and Survival Data." Annual Review of Statistics and Its Application 12 (2025): 449–476. https://doi.org/10.1146/annurev-statistics-112723-034334.

3. Wu L, Liu W, Yi GY, Huang Y. Analysis of Longitudinal and Survival Data: Joint Modeling, Inference Methods, and Issues. J Probab Stat. 2012;2012:640153. doi:10.1155/2012/640153.

4. Laird, N. M., and Ware, J. H. Random-effects models for longitudinal data. Biometrics 38 (1982), 963–974. https://doi.org/10.2307/2529876.

5. Cooper, D. J., Cai, J., and Glynn, R. J. Approaches to estimate between- and within-subject correlation coefficients in longitudinal repeated-measures studies. Journal of Statistical Computation and Simulation 83 (2013), 2091–2104. https://doi.org/10.1080/00949655.2012.687993.

6. Helbing, Dirk. Survival Analysis, Master Equation, Efficient Simulation of Path-Related Quantities, and Hidden State Concept of Transitions. arXiv preprint arXiv:cond-mat/9805361, 1998.

7. Ngwa, J. S., Cabral, H. J., Cheng, D. M., Gagnon, D. R., LaValley, M. P., & Cupples, L. A. Revisiting methods for modeling longitudinal and survival data: Framingham Heart Study. BMC Medical Research Methodology, 21 (2021), 29. https://doi.org/10.1186/s12874-021-01207-y.

8. Cheng, Y.-S., Chen, Y. and Lee, M.-L.T. "Longitudinal Survival Analysis Using First Hitting Time Threshold Regression: With Applications to Wiener Processes." Stats 8 (2025): 32. https://doi.org/10.3390/stats8020032.

9. Medina-Olivares, V., Calabrese, R., Crook, J., and Lindgren, F. Joint models for longitudinal and discrete survival data in credit scoring. European Journal of Operational Research 307 (2023), 1457–1473. https://doi.org/10.1016/j.ejor.2022.10.022.

10. Cui, E.H. A Tutorial on Statistical Models Based on Counting Processes. arXiv:2412.07114v4 [stat.AP], 4 Dec 2024.

11. Xie C, Huang X, Li R, Shen Y, Short NJ, Bhalla KN. A new cure model accounting for longitudinal data and flexible patterns of hazard ratios over time. Statistical Methods in Medical Research. 2025;34(4):683–700. doi:10.1177/09622802251320793.

12. R Core Team (2025). *R: A Language and Environment for Statistical Computing*. R Foundation for

Statistical Computing, Vienna, Austria. https://www.R-project.org/.

13. Wickham H, Averick M, Bryan J, Chang W, McGowan LD, François R, Grolemund G, Hayes A, Henry L, Hester J, Kuhn M, Pedersen TL, Miller E, Bache SM, Müller K, Ooms J, Robinson D, Seidel DP, Spinu V, Takahashi K, Vaughan D, Wilke C, Woo K, Yutani H (2019). "Welcome to the tidyverse." *Journal of Open Source Software*, *4*(43), 1686. doi:10.21105/joss.01686 https://doi.org/10.21105/joss.01686.

14. Pinheiro J, Bates D, R Core Team (2025). *nlme: Linear and Nonlinear Mixed Effects Models*. doi: 10.32614/CRAN.package.nlme https://doi.org/10.32614/CRAN.package.nlme, R package version 3.1-168, https://CRAN.R-project.org/package=nlme.

15. Therneau T (2024). *A Package for Survival Analysis in R*. R package version 3.8-3, https://CRAN.R-project.org/package=survival.

16. Rizopoulos D, Miranda Afonso P, Papageorgiou G (2025). *JMbayes2: Extended Joint Models for Longitudinal and Time-to-Event Data*. doi:10.32614/CRAN.package.JMbayes2 https://doi.org/10.32614/CRAN.package.JMbayes2, R package version 0.5-7, https://CRAN.R-project.org/package=JMbayes2.

17. Henrik Bengtsson, A Unifying Framework for Parallel and Distributed Processing in R using Futures, The R Journal (2021) 13:2, pages 208-227, doi:10.32614/RJ-2021-048.

18. Vaughan D, Dancho M (2022). *furrr: Apply Mapping Functions in Parallel using Futures*. doi: 10.32614/CRAN.package.furrr https://doi.org/10.32614/CRAN.package.furrr, R package version 0.3.1, https://CRAN.R-project.org/package=furrr.

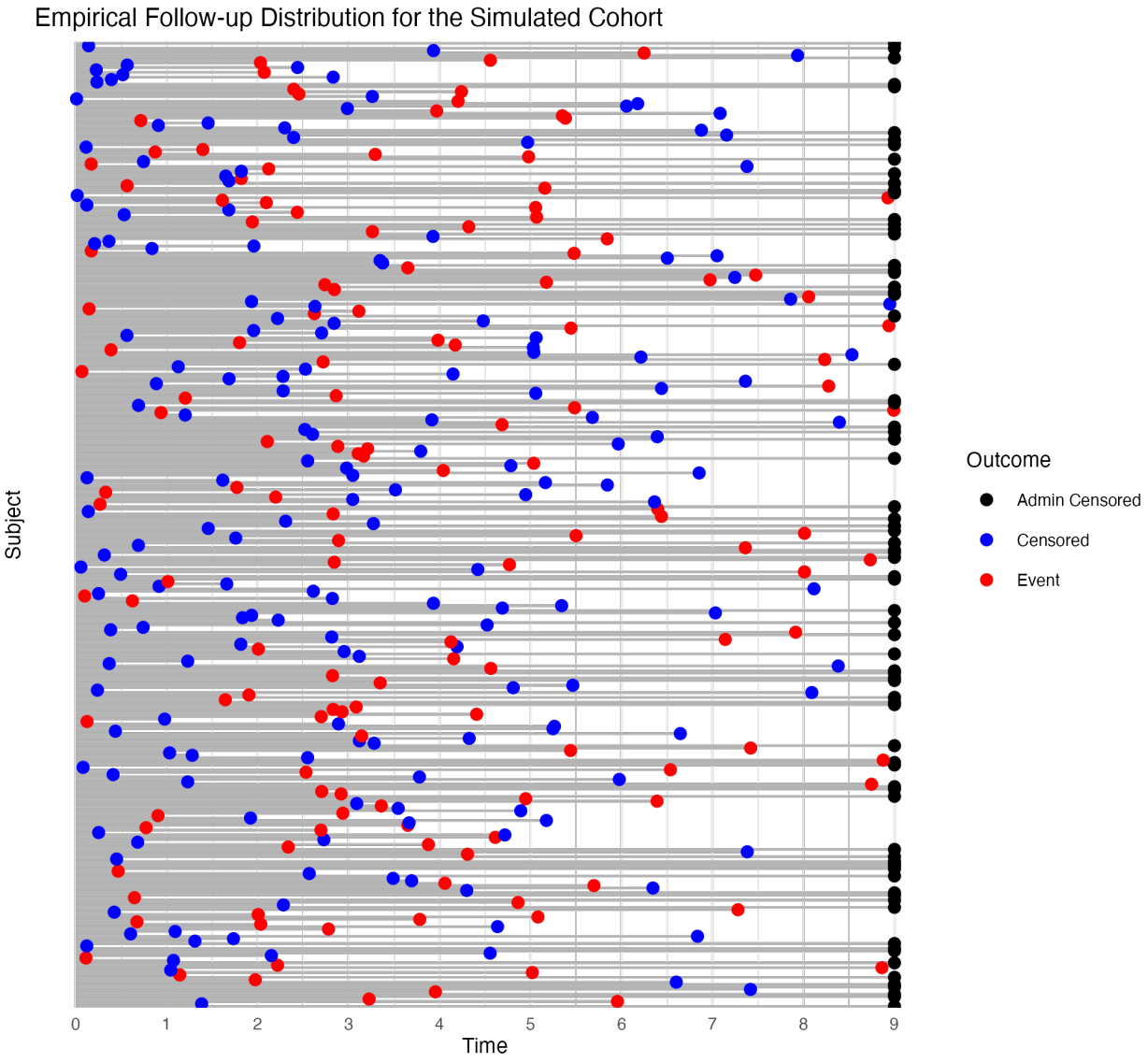
Appendix

```r
### Data Generating process

library(tidyverse)

setwd("~/Desktop/JHU Courses/Probability and Stochastic Processes I/Project/")

# Reproducibility
set.seed(12345)


generate_joint_data <- function(
  n         = 400,
  sigma_eps = 0.3,    # measurement-error SD
  alpha     = 0.5,    # longitudinal → hazard association
  rate_cens = 0.1,    # censoring rate for Exp(rate_cens)
  rho       = 0       # Box-Cox parameter: 0 = PH, 1 = PO
) {
  time_grid <- 0:9
  ## Fixed longitudinal model parameters (held constant across scenarios)
  beta_0 <- 0       # population intercept
  beta_1 <- -0.1    # population slope (slow decline)

  ## Random-effects parameters (also held fixed)
  sigma_b0 <- 1     # sd of random intercept
  sigma_b1 <- 0.3   # sd of random slope
  rho_b    <- 0.3   # corr(intercept, slope)
```

```r
n_times <- length(time_grid)
t_max   <- max(time_grid)

## Longitudinal data generation
data_long <-
  tibble(
    id   = rep(1:n, each = n_times),
    time = rep(time_grid, times = n)
  ) %>%
  arrange(id, time) %>%
  left_join(
    tibble(id = 1:n) %>%
      mutate(
        z1 = rnorm(n),
        z2 = rnorm(n),
        # (b0, b1) ~ N(0, Sigma_b)
        b0 = sigma_b0 * z1,
        b1 = rho_b * sigma_b1 * z1 +
             sqrt(1 - rho_b^2) * sigma_b1 * z2
      ) %>%
      select(id, b0, b1),
    by = "id"
  ) %>%
  mutate(
    # Latent (error-free) trajectory U_i(t)
    U_true = beta_0 + beta_1 * time + b0 + b1 * time,
    # Measurement error
    eps = rnorm(n(), mean = 0, sd = sigma_eps),
    V   = U_true + eps
  )

## Weibull baseline, calibrated so S_0(t_max) = 0.5 under the Box-Cox H_rho
kappa <- 1.3  # shape

lambda0 <- if (abs(rho) < 1e-8) {
  # PH case: H_0(u) = u -> Lambda_0(t) = lambda t^kappa -> S_0(t) = exp(-lambda t^kappa)
  log(2) / (t_max^kappa)
} else {
  # Box-Cox: H_rho(u) = log(1 + rho u)/rho, want H_rho(lambda t_max^kappa) = log 2
  # -> log(1 + rho lambda t_max^kappa)/rho = log 2 -> lambda = (2^rho - 1) / (rho t_max^kappa)
  (2^rho - 1) / (rho * t_max^kappa)
}

# Baseline hazard lambda_0(t)
h0 <- function(t) {
  kappa * lambda0 * t^(kappa - 1)
}

# Latent trajectory as function of continuous time
U_fun <- function(t, b0, b1) {
  beta_0 + beta_1 * t + b0 + b1 * t
}
```

```r
# Subject-specific "raw" cumulative hazard A_i(t) = integral lambda_0(s) exp{alpha U_i(s)} ds
A_fun <- function(t, b0, b1) {
  if (t <= 0) return(0)
  integrate(
    f     = function(s) h0(s) * exp(alpha * U_fun(t = s, b0, b1)),
    lower = 0,
    upper = t
  )$value
}

# Transformed cumulative hazard lambda_i(t) = H_rho(A_i(t))
Lambda_fun <- function(t, b0, b1) {
  A_val <- A_fun(t, b0, b1)
  if (abs(rho) < 1e-8) {
    A_val                       # PH: H_0(u) = u
  } else {
    log(1 + rho * A_val) / rho # Box-Cox transform
  }
}

# Inverse transform sampling: solve lambda_i(T_i) = -log W_i
sim_T <- function(b0, b1, t_cap = 20) {
  W_i <- runif(1)
  c_i <- -log(W_i)
  f_i <- function(t) Lambda_fun(t, b0, b1) - c_i

  # Cumulative hazard at a large cap time
  Lambda_cap <- Lambda_fun(t_cap, b0, b1)

  # If even by t_cap we have not reached c_i, treat event as beyond t_cap
  if (Lambda_cap <= c_i) {
    return(t_cap)
  }

  # Otherwise there is a unique root in (0, t_cap)
  uniroot(f_i, interval = c(1e-6, t_cap))$root
}

## Simulate event times and censoring
subjects <-
  data_long %>%
  distinct(id, b0, b1) %>%
  rowwise() %>%
  mutate(
    # True event time from transformation model
    T_true = sim_T(b0, b1),

    # Independent censoring C_i ~ Exp(rate_cens)
    C = rexp(1, rate = rate_cens),

    # Observed time and event indicator within [0, t_max]
    Y     = pmin(T_true, C, t_max),
    delta = as.integer(T_true <= C & T_true <= t_max)
```

```
    ) %>%
    ungroup() %>%
    select(id, b0, b1, T_true, C, Y, delta)

  ## Truncate longitudinal process at Y_i
  data_long_joint <-
    data_long %>%
    left_join(subjects, by = c("id", "b0", "b1")) %>%
    filter(time <= Y)

  list(
    data_joint_long = data_long_joint,  # truncated longitudinal data
    surv = subjects,                # subject-level event/censoring
    args = list(
      n         = n,
      sigma_eps = sigma_eps,
      alpha     = alpha,
      rate_cens = rate_cens,
      rho       = rho
    )
  )
}
```



Observed follow-up times for all simulated subjects. Horizontal grey segments show each subject's observed longitudinal window from studyentry to their event or censoring time within the administrative horizon t = 9. Points indicate observed outcomes: Red: events, Blue: censoring before t=9, Black: administrative ensoring at t=9. All subjects are displayed beginning at t = 0 for demonstration purposes. Delayed entry (left truncation) was not considered in this simulation.

```
### Fit Estimators

library(nlme)
library(survival)
library(JMbayes2)

fit_estimators <- function(sim_obj) {
  ## Unpack simulated data
  long_dat <- sim_obj$data_joint_long  # longitudinal, truncated at Y
  surv_dat <- sim_obj$surv             # subject-level survival info: id, Y, delta, ...

  ## 1. Stage-1 LMM (used by both estimators)
  # Random intercept + slope on time
  lme_fit <- try(
```

```
  nlme::lme(
    V ~ time,
    random   = ~ time | id,
    data     = long_dat,
    control  = nlme::lmeControl(
      msMaxIter = 200,
      msMaxEval = 200
    )
  ),
  silent = TRUE
)

if (inherits(lme_fit, "try-error")) {
  # If the mixed model fails, nothing else is usable for this replicate
  return(tibble::tibble(
    alpha_joint   = NA_real_,
    alpha_two     = NA_real_,
    beta0         = NA_real_,
    beta1         = NA_real_,
    sigma_eps_hat = NA_real_,
    var_b0        = NA_real_,
    var_b1        = NA_real_,
    cov_b0b1      = NA_real_
  ))
}

# Extract longitudinal parameters from lme
beta_hat      <- nlme::fixef(lme_fit)
D_hat         <- as.matrix(nlme::getVarCov(lme_fit))  # 2x2 random-effects covariance
sigma_eps_hat <- lme_fit$sigma

## 2. Two-stage estimator: Cox with plug-in U_hat(t)
# Build counting-process style data with time-dependent covariate U_hat(t)
long_for_cox <-
  long_dat %>%
  select(id, time, V) %>%
  left_join(
    surv_dat %>% select(id, Y, delta),
    by = "id"
  ) %>%
  group_by(id) %>%
  arrange(time, .by_group = TRUE) %>%
  mutate(
    start = time,
    stop  = lead(time),
    # Last interval goes to min(last grid time, Y)
    stop  = ifelse(is.na(stop) | stop > Y, Y, stop)
  ) %>%
  ungroup() %>%
  filter(start < Y) %>%  # remove any degenerate intervals
  mutate(
    # Event occurs at the interval that ends at Y and delta == 1
    event = as.integer(stop == Y & delta == 1)
```

```
  )

# Predicted subject-specific trajectories (current value) at the start of each interval
long_for_cox$U_hat <- as.numeric(
  predict(lme_fit, newdata = long_for_cox, level = 1)
)

# Cox PH with time-dependent internal covariate (plug-in)
cox_two <- try(
coxph(Surv(start, stop, event) ~ U_hat, data = long_for_cox),
silent = TRUE
)

alpha_two <- if (inherits(cox_two, "try-error")) NA_real_ else coef(cox_two)[["U_hat"]]

## 3. Joint model estimator via JMbayes2
# Baseline Cox model with no external covariates
cox_base <- try(
  coxph(
    Surv(Y, delta) ~ 1,
    data  = surv_dat,
    x     = TRUE,
    model = TRUE
  ),
  silent = TRUE
)

if (inherits(cox_base, "try-error")) {
  # If baseline Cox fails (e.g. no events), joint model cannot be fit
  alpha_joint <- NA_real_
} else {
  # Joint model: Surv_object = cox_base, Mixed_objects = list(lme_fit)
  jm_fit <- try(
    JMbayes2::jm(
      Surv_object   = cox_base,
      Mixed_objects = list(lme_fit),
      time_var      = "time",
      n_iter        = 4000,
      n_burnin      = 1000,
      n_thin        = 5
    ),
    silent = TRUE
  )

  if (inherits(jm_fit, "try-error")) {
    alpha_joint <- NA_real_
  } else {
    alpha_joint <- as.numeric(JMbayes2::coef(jm_fit))
  }
}

## 4. Collect and return estimates
tibble(
```

```
  alpha_joint   = alpha_joint,
  alpha_two     = alpha_two,
  beta0         = beta_hat["(Intercept)"],
  beta1         = beta_hat["time"],
  sigma_eps_hat = sigma_eps_hat,
  var_b0        = D_hat[1, 1],
  var_b1        = D_hat[2, 2],
  cov_b0b1      = D_hat[1, 2]
)
}
```

```
### Parallel Monte Carlo

library(future)
library(furrr)

# Use all but one core
plan(multisession, workers = max(1, parallel::detectCores() - 1))

# For reproducible parallel RNG
RNGkind("L'Ecuyer-CMRG")

run_sim_scenario <- function(
  n_sims    = 1000,
  n         = 400,
  sigma_eps = 0.3,
  alpha     = 0.5,
  rate_cens = 0.1,
  rho       = 0
) {
  furrr::future_map_dfr(
    1:n_sims,
    function(rep_idx) {
      # 1. Generate one dataset under the DGP
      sim_dat <- generate_joint_data(
        n         = n,
        sigma_eps = sigma_eps,
        alpha     = alpha,
        rate_cens = rate_cens,
        rho       = rho
      )

      # 2. Fit joint and two-stage estimators
      est_tab <- fit_estimators(sim_dat)

      # 3. Attach replication index
      est_tab %>%
        mutate(rep = rep_idx, .before = 1)
    },
    .options = furrr::furrr_options(seed = TRUE)  # reproducible across workers
  )
}
```

```r
## Simulation grid

sim_grid <- expand.grid(
  sigma_eps = c(0.30, 0.75, 1.50),
  alpha     = c(0.25, 0.50, 1.00),
  rate_cens = c(0.05, 0.10, 0.25),
  rho       = c(0,1)  # misspecification check
) %>% tibble::as_tibble()


(start <- Sys.time())
results <- purrr::map_df(1:nrow(sim_grid), function(i) {
  pars <- sim_grid[i, ]
  out <- run_sim_scenario(
    n_sims    = 30,
    n         = 400,
    sigma_eps = pars$sigma_eps,
    alpha     = pars$alpha,
    rate_cens = pars$rate_cens,
    rho       = pars$rho
  )
  out$scenario_id <- i
  bind_cols(pars, out)
})
end <- Sys.time()
end-start

results

saveRDS(results, file = "simulation_results.rds")
```