JOINT VARIABLE SELECTION IN PROTEOMICS SURVIVAL MODELS

Jakub Bajzik*, **Al Depope***, **Marco Mondelli**† **& Matthew R. Robinson**†
Institute of Science and Technology Austria
Am Campus 1, 3400 Klosterneuburg, Austria
{jakub.bajzik, al.depope, marco.mondelli, matthew.robinson}@ist.ac.at

Yasaman Zolfimoselo & Alexander Sharipov
École Polytechnique Fédérale de Lausanne
Bâtiment CE – 3.316 Station 1, CH – 1015 Lausanne, Switzerland
{yasaman.zolfimoselo, alexander.sharipov}@epfl.ch

ABSTRACT

The incidence of the vast majority of neurodegeneration, cancer, and metabolic diseases generally increases exponentially with age. In large-scale biobanks, linking time-to-diagnosis information in electronic health records to multiple genomic ("multiomics") measures has the potential to reveal the genes and biological pathways involved in the disease onset and progression. To date, association testing has commonly been conducted by testing one variable at a time using semi-parametric Cox proportional hazards (CoxPH) models, which ignores correlation structure and increases the risk of false discoveries. To address these issues, we introduce a novel fully parametric computational method, vampW, based on the Bayesian Vector Approximate Message Passing framework applied to a Weibull model. vampW jointly models correlated features, while providing an interpretable hazard structure, producing a continuous survival curve, and incorporating prior knowledge. In an extensive simulation study, we demonstrate that joint modeling of proteomics data and time-to-event outcomes using vampW substantially reduces false discoveries in comparison to marginal testing and other forms of joint CoxPH models. The application of vampW to 2,924 proteins across 24 diseases in 53,018 individuals from the UK Biobank, identifies 219 protein associations, the majority of which are not among the top marginal discoveries. vampW also achieves a significant improvement in the prediction of disease onset times: across 14 medical outcomes, it reduces the root mean squared error by over 32% and 26%, when compared respectively to CoxPH variants and the deep learning approach DeepSurv. In addition, vampW outperforms deep learning methods in the data-scarce regime on common survival benchmarking datasets. In summary, vampW offers accurate and interpretable variable selection and out-of-sample prediction within a single computational framework, making it a powerful tool for dissecting the proteomic architecture of human health span.

1 INTRODUCTION

Aging is the primary risk factor for the vast majority of common chronic, noncommunicable diseases that cause illness and death worldwide. The relationship is generally exponential, where disease risk and incidence increase sharply with advancing age (Kopp, 2024). The potential genes and pathways involved in disease onset and progression can be identified by associating time-to-diagnosis information contained within electronic health records to multiple genomic ("multiomics") measures (Gadd et al., 2024; Sun et al., 2023). The prognostic biomarkers identified can then provide predictors of

*Equal contribution.
†Joint supervision.

age at disease onset, facilitating patient screening programs and preventative medicine (Gadd et al., 2024; Sun et al., 2023).

To date, the two main tasks in survival modeling of multiomics data—association testing and out-of-sample prediction of disease onset age—are almost always conducted using semiparametric Cox proportional hazards (PH) models. More precisely, association testing of disease onset age in genomic studies has predominantly been conducted with marginal CoxPH testing (Gadd et al., 2024; Gong et al., 2025; Deng et al., 2025). This corresponds to analyzing one genomic feature at a time and it ignores pervasive correlations among genomic measures (Argentieri et al., 2024), making it difficult to identify which proteins are most likely driving the observed patterns, with an increased risk of false discoveries (Fan et al., 2024). For prediction tasks, penalized CoxPH models (Tibshirani, 1997) are frequently employed (Gadd et al., 2024; Kuo et al., 2025), enabling the simultaneous modeling of many markers while shrinking or zeroing out a subset of coefficients. Cross-validation is commonly used to select the optimal level of penalization. Stability selection (Meinshausen & Bühlmann, 2010) can further be applied to identify the most robust predictors and reduce the impact of stochastic variability in model fitting. For both tasks (namely, association testing and prediction), the CoxPH model relies on a partial likelihood that ignores the baseline hazard and the exact spacing between events. While this makes semi-parametric Cox regression robust and widely applicable, it is statistically less efficient because it depends only on the ordering of event times rather than their absolute values. Hence, it does not provide direct predictions of absolute event times. Instead, it yields relative risk scores with respect to other individuals in the dataset.

Here, we address the limitations of semi-parametric models and the false discoveries of marginal testing by introducing vampW, a joint estimation procedure based on Vector Approximate Message Passing (VAMP) (Rangan et al., 2019; Schniter et al., 2016) that employs a fully parametric Weibull survival model. Approximate message passing (AMP) (Donoho et al., 2009; Feng et al., 2022) refers to a family of iterative algorithms with several attractive properties for the proposed application, namely *(i)* allowing the use of a wide range of Bayesian priors, *(ii)* achieving Bayes-optimal performance in several settings (Barbier et al., 2019; Montanari & Venkataramanan, 2021; Mondelli & Venkataramanan, 2021; Barbier et al., 2023), and *(iii)* enabling association test statistics to be obtained via state evolution (Bayati & Montanari, 2011; Bolthausen, 2014), a low-dimensional recursion tracking the probability distributions of objects inside the algorithm. (Depope et al., 2026) have demonstrated the applicability of the AMP framework to genome-wide association studies (GWAS) and whole genome sequence data (WGS). However, the model considered by (Depope et al., 2026) is linear and the data is not censored, as is standard in GWAS and WGS. Here, in contrast, we analyze protein measures from the UK Biobank Pharma Proteomics Project (PPP) dataset (Sun et al., 2023), where the majority of the data is right-censored, see Section 2.1 for details. Furthermore, the Weibull survival model we employ, whose applicability to omics and medical data is well documented in the literature (Ojavee et al., 2021; Plana et al., 2022; Duan et al., 2018), is highly non-linear. The methodological novelty of vampW lies in the introduction of Weibull channel denoisers and in handling right-censoring within VAMP framework. The key differences between vampW and the marginal CoxPH method are outlined below:

- In the vampW framework, all markers are modeled jointly. This allows the modeling of correlated features, and it captures combined effects that marginal one-at-a-time tests cannot detect.
- In vampW, the markers are modeled under the Weibull model, which offers an interpretable hazard structure. Depending on the value of its shape parameter, the hazard rate can increase, decrease, or remain constant over time.
- vampW is based on the full likelihood, which captures the probability of the observed data under the complete model specification. In contrast to the partial likelihood used by CoxPH models, the full-likelihood approach exploits the actual event times directly.
- vampW produces a continuous survival curve, whereas the Cox model paired with the Breslow estimator yields a piecewise-defined baseline hazard. As a result, the Cox model cannot extrapolate beyond the last observed event time, while vampW can generate long-term survival predictions because its parametric form is defined over the full time domain.
- vampW naturally supports the incorporation of prior knowledge or specific information within a Bayesian framework. The method produces posterior mean estimates together with credible intervals, allowing for explicit quantification of uncertainty.

Using large-scale proteomics data linked to electronic health records (Sun et al., 2023; Hu et al., 2025; Wang et al., 2025), we conduct extensive simulations and benchmarking analyses to demonstrate that joint modeling of time-to-event outcomes using vampW substantially reduces false-positive protein associations, as compared to conventional one-protein-at-a-time testing and other forms of CoxPH models. Across the 24 outcomes, we discover 219 protein associations and show that protein–outcome associations are sensitive to the method of age correction. In addition to identifying several well-established biomarkers, we find novel associations that may point to previously unrecognized disease pathways. In particular, we identify *CXCL17*, previously reported as a potential biomarker for colorectal cancer and ischemic heart disease, as being associated with depression. Moreover, across 14 medical outcomes with sufficient prevalence for out-of-sample evaluation, vampW predicts age at onset with an average root mean squared error which is over 32% and 26% lower than CoxPH variants and the deep learning approach DeepSurv (Katzman et al., 2018), respectively. Finally, we show that vampW outperforms deep learning methods in the data-scarce regimes on standard survival benchmarking datasets, demonstrating the utility of our approach across datasets in the life sciences.

2 Data modeling and existing methods

2.1 Weibull model for time-to-event outcomes

We consider a Weibull(α, η) model, one of the most common models in survival analysis, where α and η are the shape and scale parameters. We follow a recently proposed formulation of the model (Ojavee et al., 2021), where we aim to estimate mean and variance of the time-to-event outcome. Let y_i be the time at which the individual i experiences the event, and consider the survival function $S(y_i) = \exp(-(\frac{y_i}{\eta_i})^\alpha)$. In the logarithmic space, the mean and variance of the model can be separated as follows:

$$\mathbb{E}[\log(y_i)] = \log(\eta_i) - \frac{K}{\alpha}, \tag{1}$$

$$\text{Var}[\log(y_i)] = \frac{\pi^2}{6\alpha^2}, \tag{2}$$

where K is the Euler-Mascheroni constant (≈ 0.57722). Once the variance is independent of the scale parameter η_i, one can introduce the vector of underlying latent genetic effects $\boldsymbol{\beta}$ as $\boldsymbol{\eta} = \exp(\mu + \boldsymbol{X}\boldsymbol{\beta} + \frac{K}{\alpha})$, resulting in $\mathbb{E}[\log(\boldsymbol{y})] = \mu + \boldsymbol{X}\boldsymbol{\beta}$, where $\boldsymbol{X} \in \mathbb{R}^{N\times P}$ is a design matrix containing observations on N individuals and P biological markers, $\boldsymbol{y} = [y_1, \ldots, y_N]$, $\boldsymbol{\eta} = [\eta_1, \ldots, \eta_N]$, $\boldsymbol{\beta}$ is the vector of marker effects, and μ represents the model intercept. The columns of the matrix $\boldsymbol{X}$ are normalized, i.e., they each have zero mean and unit variance. Using the fact that the logarithm of a Weibull random variable follows a Gumbel distribution and the Location-Scale property of the Gumbel distribution, we formulate the final model as

$$\log(\boldsymbol{y}) = \mu + \boldsymbol{X}\boldsymbol{\beta} + \frac{\boldsymbol{w}}{\alpha} + \frac{K}{\alpha}, \tag{3}$$

where $\boldsymbol{w} = [w_1, \ldots, w_N]$ with each w_i following the Gumbel$(0, 1)$ distribution. We note that in the algorithm and software described below, any arbitrary measurement represented in floating point can be accommodated as the entries of $\boldsymbol{X}$. These can be gene expression, methylation, copy number variants, DNA sequence variation, protein measures, which can all be included alongside one another.

To allow for a range of effect sizes from different data types, we select the prior on the marker effects $\boldsymbol{\beta}$ to be distributed as a form of spike and slab distribution, with non-zero signals coming from a mixture of normal distributions,

$$\beta_j \sim (1-\lambda)\delta_0 + \lambda \sum_{i=1}^{L} \pi_i \cdot \mathcal{N}(0, \sigma_i^2), \tag{4}$$

where $(\pi_1, \ldots, \pi_L)$ denotes the vector of inclusion probabilities for each of the L mixture components, δ_0 is a dirac spike function at 0, λ is the sparsity ratio, and $(\sigma_1^2, \ldots, \sigma_L^2)$ are mixture-specific variances.

Finally, we account for the right-censoring issue inherent in time-to-event data. In practical applications, the true event time y_i is often right-censored, meaning that the event of interest has not occurred for some individuals by the end of the study. Consequently, we do not observe y_i directly for all i; instead, we observe the time $\min(y_i, C_i)$ and an indicator $\delta_i = \mathbb{I}(y_i \leq C_i)$, where C_i is the censoring time and $\delta_i = 1$ implies the event was observed while $\delta_i = 0$ implies it was censored.

2.2 UK BIOBANK PROTEIN DATA AND TIME-TO-EVENT PHENOTYPES

To validate our proposed approach, we analyze protein measures from the UK Biobank Pharma Proteomics Project (PPP) dataset (Sun et al., 2023). We use a total of 2,924 unique proteins measured across 53,018 individuals, obtained using the antibody-based Olink Explore 3072 PEA. We standardize each protein vector across individuals (ignoring missing values) to obtain zero mean and unit variance, and then impute the missing values with zeros. We adjust the protein levels by regressing them on the covariates sex and age via the linear model $\alpha_1 \cdot \mathrm{sex} + \alpha_2 \cdot \mathrm{age}$. We then use the resulting residuals, representing the component of protein variation not explained by sex and age. This follows previous studies which use the same correction factors (Deng et al., 2025; Gadd et al., 2024), having shown that adjusting for genetic principal components, protein batch, and study center has minimal effects on protein levels. Additionally, to explore non-linear effects of age on protein levels (Sigurdsson et al., 2025), which have not been previously considered in protein-disease onset association testing, we perform a covariate adjustment by including an exponential age component, alongside the linear term, in the model $\alpha_1 \cdot \mathrm{sex} + \alpha_2 \cdot \mathrm{age} + \alpha_3 \cdot \exp(\mathrm{age})$. Throughout all analyses, we use a 90%/10% split for train and test subsets.

The 24 phenotypes under investigation are described in Table 1 deferred to Appendix A. We choose not to include schizophrenia because preliminary analyses of its age-at-onset distribution reveal a substantial deviation from a Weibull model, along with a comparatively uniform hazard rate across the observed time span, see Figure 4 in Appendix B. We consider an individual's time of event from the UK Biobank health records of the first reported occurrence (Category 1712). For major depression, we combine two data fields into one joint set, particularly the first reported depressive episode (UK Biobank code 130894) and the first reported recurrent depressive disorder (UK Biobank code 130896). Similarly, for inflammatory bowel disease, we combine the first reported cases of Crohn's disease (UK Biobank code 131626) and ulcerative colitis (UK Biobank code 131628) into one joint set. In the type 2 diabetes cohort, we exclude individuals who are observed to have type 1 diabetes (UK Biobank code 130706) at the same time.

For cancer phenotypes (Breast, Prostate, Colorectal, Lung, Gynecological, and Brain/CNS), we use data from the UK Biobank cancer registry. Cases were defined by the presence of disease-specific ICD-10 (field 40006) or ICD-9 (field 40013) codes across any of the 22 reported instances. The specific codes for each cancer type are detailed in Table 2 in Appendix A. For identified cases, the time-to-event was defined as the minimum age at diagnosis recorded across all matching instances (field 40008). For individuals without a diagnosis (censored observations), the time-to-event was calculated as their age at the initial assessment center visit, derived from the date of assessment (field 53) and year of birth (field 34).

2.3 SIMULATION STUDY

We conduct a simulation study to demonstrate the utility of the vampW method as a prediction and association-testing tool. We exploit pre-processed (standardized, not adjusted for covariates) real protein measures from the UK Biobank PPP dataset, described above. We simulate artificial phenotypic outcomes in different scenarios, varying multiple parameters, namely: *(i)* the proportion of variance explained by biological markers $\sigma^2_G \in \{40\%, 80\%\}$, *(ii)* the proportion of censored individuals $C \in \{90\%, 95\%\}$, *(iii)* the phenotype distribution (Weibull, ExpGamma), *(iv)* the prior distribution of the slab part of the regression coefficients (Normal, Laplace), and *(v)* the ExpGamma distribution parameter $\kappa \in \{0.8, 1.2\}$. This yields 24 scenarios in total, summarized in Table 3 in Appendix A. Details on simulating ExpGamma outcomes are in Appendix F.

The causal signals are sampled from either an i.i.d. zero-mean Bernoulli-Gaussian or Bernoulli-Laplace spike and slab distribution:

$$\beta_j \sim (1-\lambda)\delta_0 + \lambda \cdot \mathcal{N}\left(0, \frac{\sigma_G^2}{P_C}\right), \tag{5}$$

$$\beta_j \sim (1-\lambda)\delta_0 + \lambda \cdot \text{Laplace}\left(0, \sqrt{\frac{\sigma_G^2}{2P_C}}\right), \tag{6}$$

where σ_G^2 is the target variance explained by biological markers (plasma protein levels). The aim is to spread the variance explained to a certain number of causal markers P_C. We use $\lambda = 0.1$ which, given the total of 2,924 markers, corresponds to roughly 292 causal markers. Note that the variance of the Laplace distribution is determined by its scale parameter. We then generate the phenotypic outcome y_i as either a Weibull or an ExpGamma distribution, as described in Appendix F.

To simulate data that more closely mimics real-world cohorts, we also account for censoring in the simulation procedure. Specifically, we randomly select a proportion C of individuals to be censored. For each censored individual i, we multiply the simulated outcome y_i with a random factor drawn from a uniform distribution $\text{Unif}(0, 1)$. We repeat the simulation 20 times for each scenario to obtain independent realizations of the simulated data. We also resample train/test sub-splits in ratio $90\%/10\%$ across data realizations.

2.4 Benchmarking methods

We evaluate vampW against state-of-the-art survival analysis approaches, specifically: *(i)* one-at-a-time marginal CoxPH testing (CoxPH marginal); *(ii)* non-penalized CoxPH fitted to all markers simultaneously (CoxPH); *(iii)* LASSO-penalized CoxPH (Tibshirani, 1997) with cross-validation and stability selection (CoxPH LASSO); and *(iv)* DeepSurv (Katzman et al., 2018). One-protein-at-a-time testing (CoxPH marginal) is among the most commonly used approaches for variable selection. Joint models offer an alternative which fit each marker in the structural context of all other markers. Furthermore, a penalized model (CoxPH LASSO) enables joint selection of markers while shrinking or zeroing out some coefficients. Cross-validation is used to determine the optimal level of penalization, and stability selection is then applied to identify the most robust predictors and reduce the impact of stochastic variability. DeepSurv represents one of the most commonly used deep learning approaches for survival analysis, capable of modeling complex nonlinear relationships among biological markers. For CoxPH and CoxPH marginal, we use the Python *lifelines* (Davidson-Pilon, 2019) library with its default settings, which estimates the baseline hazard using the Breslow estimator and outputs the regression coefficients and corresponding p-values. Bonferroni correction is applied for multiple testing rejecting the null hypothesis H_0 if the p-value $p < 0.05/P$, where $P = 2{,}924$ tests correspond to the number of proteins.

For traits where CoxPH with default parameters failed to converge, we follow the guidelines provided by the *lifelines* documentation for resolving convergence issues in the Cox Proportional Hazard model[1]. Consistent with these recommendations, we use fixed initialization parameters with adjusted penalization and step sizes to ensure stability. Specifically, for liver disease, we employ a penalization of 0.1 and a step size of 0.95; for Alzheimer's disease, multiple sclerosis, Parkinson's disease, systemic lupus erythematosus, vascular dementia, as well as the colorectal, gynecological, and lung cancer datasets, we employ a penalization of 0.001 with a step size of 0.95; finally, for inflammatory bowel disease, and brain and CNS cancer, we employ a penalization of 0.001 and a step size of 0.5.

For CoxPH LASSO, we use Python's *scikit-survival* (Pölsterl, 2020) implementation with a grid search to identify the optimal penalty. We perform 5-fold cross-validation, further splitting the training data, and sweep over 16 penalizer values in logarithmic space, from 10^{-4} to 10^{0}. After selecting the optimal penalizer, we perform stability selection (Meinshausen & Bühlmann, 2010) by fitting 100 parallel penalized regressions, each trained on a random 75% subset of the data. We then retain only the stable variables appearing in at least 90% of the models. Finally, we fit a non-penalized CoxPH model from *lifelines* using only these stable features, estimating the baseline hazard via the Breslow method to obtain the final refined regression coefficients and their corresponding p-values.

[1] `https://lifelines.readthedocs.io/en/latest/Examples.html#problems-with-convergence-in-the-cox-proportional-hazard-model`

For running DeepSurv, we use the code provided by the authors (Katzman et al., 2018), using one hidden layer and turning the batch normalization and standardization on. To optimize performance, we explore the hyperparameter space via a grid search over the following parameters: $\ell_2\text{Reg} \in \{0.01, 1, 5\}$, $\text{dropout} \in \{0, 0.1\}$, learning rate (LR) $\in \{0.001, 0.0001\}$, size of the hidden layer $\in \{256, 512, 1024\}$. Moreover, we fix the LR decay to 0.001, momentum for the Adam optimizer to 0.9 and number of epochs to 500. During the grid search, the training set is further randomly split into an 90% sub-train set and a 10% validation subset, optimizing for the best C-index on the validation set. To predict age at onset using DeepSurv, we use the predicted log-risk scores from DeepSurv as input to the *lifelines* CoxPHFitter, which estimates the baseline hazard using the Breslow estimator and predicts expected onset times. For CoxPHFitter, we set the step size to 0.5.

All benchmarking methods described here, including vampW, operate on the outcome vector $\boldsymbol{y}$ in the time domain, after standardization on the logarithmic scale, so that $\log(\boldsymbol{y})$ has zero mean and unit variance. To obtain root mean square error (RMSE) values in units of years for predictions of UK Biobank outcomes, we rescale the model predictions from the log-standardized space back to the original time scale, using the mean and standard deviation of $\log(\boldsymbol{y}_{train})$, with $\boldsymbol{y}_{train}$ denoting the training part of $\boldsymbol{y}$. The RMSE is evaluated using only uncensored samples, as the true event times for censored data are unobserved and the benchmarked methods do not explicitly model the censoring time distribution.

3 VAMPW ALGORITHM AND ITS IMPLEMENTATION

Our proposed algorithm, vampW, is based on Vector Approximate Message Passing (VAMP) for generalized linear models (Rangan et al., 2019), (Schniter et al., 2016), where the outcome variable is observed through a noisy and non-linear Weibull channel. It adaptively estimates the prior-specific parameters $\Theta = \{\lambda, (\pi_i)_{i=1}^{L}, (\sigma_i^2)_{i=1}^{L}\}$ via a Bayesian Expectation-Maximization procedure, as introduced in EM-VAMP (Fletcher & Schniter, 2017). vampW introduces a novel approach for handling censored data, as well as denoisers derived specifically for Weibull link functions and the corresponding expectation-maximization updates. We highlight that no existing VAMP method is able to address the challenge of censored data.

As seen in Algorithm 1, vampW consists of two denoising steps and two (non-linear) minimum mean square error (MMSE) estimation steps, applied separately to the vector of effect sizes $\boldsymbol{\beta}$ and to the latent predictor. The denoising step accounts for the prior properties of the model when considering a noisy estimate of either $\boldsymbol{\beta}$ or the latent predictor, while the MMSE step further refines the estimates by taking the correlation structure among the columns in the design matrix $\boldsymbol{X}$ into account. The key feature is the so-called *Onsager correction*, which is added to ensure the asymptotic normality, of the noise corrupting the algorithm's estimates of $\boldsymbol{\beta}$ at every iteration, important for the subsequent feature selection. A direct application of generalized EM-VAMP to real genomics, epigenetics and proteomics measures may lead to signal-estimation divergence, primarily due to violations of right-rotational invariance in the data design matrix, one of the assumptions in the original EM-VAMP algorithm. To mitigate these stability issues, we perform *damping* of denoised signals following (Rangan et al., 2019), as seen on line 11 of Algorithm 1, where $\rho \in (0, 1)$ denotes the damping factor. This approach makes the algorithm take smaller steps when learning $\boldsymbol{\beta}$. In order to further stabilize convergence and improve prediction performance, we employ damping autotuning (Vila et al., 2015): we iteratively reduce the damping by the factor c_ρ while the training prediction, using signal estimates from the denoising step $\hat{\boldsymbol{\beta}}_1$, increases over the last iteration, measured by the concordance index (C-index), as seen in line 12 of Algorithm 1. The damping choices affect the convergence properties when searching over the parameter space, as too strong damping may cause the algorithm to get stuck in local minima; and thus we propose a grid search of starting parameters which is facilitated by the efficient implementation of our algorithm. Following (Depope et al., 2026), we employ the Hutchinson estimator (Hutchinson, 1990) to evaluate the trace of the inverse of regularized Gram matrices, and we warm-start the conjugate gradient (CG) algorithm approximating the solution of the linear system (line 32 in Algorithm 1). The CG solver is initialized from the value at convergence in the previous iteration, which leads to reduced computational time. Compared to the gVAMP algorithm introduced by (Depope et al., 2026) for a linear model without censoring, vampW derives a non-linear Weibull model within the AMP framework and incorporates censoring information into the process (as detailed in the paragraph below), thus requiring two denoising steps and two non-linear MMSE estimation steps.

Algorithm 1 vampW

Input: max number of iterations N_{it},
normalized design data matrix of uncensored individuals $\boldsymbol{X} \in \mathbb{R}^{N \times P}$,
normalized design data matrix of censored individuals $\boldsymbol{X}_{\mathcal{C}} \in \mathbb{R}^{N_{\mathcal{C}} \times P}$,
vector of measured phenotypes $\boldsymbol{y} \in \mathbb{R}^{N}$,
vector of censoring times $\boldsymbol{y}_{\mathcal{C}} \in \mathbb{R}^{N_{\mathcal{C}}}$,
initial noisy signal estimate $\boldsymbol{r}_{10} \in \mathbb{R}^{P}$,
initial signal precision estimate $\gamma_{10} > 0$,
initial noisy genetic predictor estimate $\boldsymbol{p}_{10} \in \mathbb{R}^{N}$,
initial genetic predictor precision estimate $\tau_{10} > 0$,
initial set of parameters defining prior distribution Θ_0,
initial estimate of Weibull model hyperparameters $\mu \in \mathbb{R}$ and $\alpha > 0$,
damping factor ρ,
damping reduction factor $c_\rho < 1$,
max damping autotune steps N_{ρ_steps},
flag to enable damping autotuning
for $t = 0, 1, \ldots, N_{\text{it}}$ **do**
  Denoising of the effect sizes
  EM update of the prior distribution parameters Θ, called Θ_t
  $\hat{\boldsymbol{\beta}}_{\text{new}} = \mathbb{E}[\boldsymbol{\beta} | \boldsymbol{r}_{1t} = \boldsymbol{\beta} + \mathcal{N}(0, \gamma_{1t}^{-1}\boldsymbol{I}), \gamma_{1t}, \Theta_t]$
  if $t = 0$ **then**
    $\hat{\boldsymbol{\beta}}_{1t} = \hat{\boldsymbol{\beta}}_{\text{new}}$
  else
    if damping autotuning is enabled **then**
      for $k = 1, \ldots, N_{\rho\_steps}$ **do**
        $\hat{\boldsymbol{\beta}}_{1t} = \rho \cdot \hat{\boldsymbol{\beta}}_{\text{new}} + (1 - \rho) \cdot \hat{\boldsymbol{\beta}}_{1t-1}$
        if C-index$_{\textbf{train}}(\hat{\boldsymbol{\beta}}_{1t}) \geq$ C-index$_{\textbf{train}}(\hat{\boldsymbol{\beta}}_{1t-1})$ **then**
          break
        else
          $\rho \leftarrow \rho \cdot c_\rho$
        end if
      end for
    else
      $\hat{\boldsymbol{\beta}}_{1t} = \rho \cdot \hat{\boldsymbol{\beta}}_{\text{new}} + (1 - \rho) \cdot \hat{\boldsymbol{\beta}}_{1t-1}$
    end if
  end if
  $\alpha_{1t} = \gamma_{1t} \cdot \langle \text{Var}[\boldsymbol{\beta} | \boldsymbol{r}_{1t} = \boldsymbol{\beta} + \mathcal{N}(0, \gamma_{1t}^{-1}\boldsymbol{I}), \gamma_{1t}, \Theta_t] \rangle$
  $\gamma_{2t} = \gamma_{1t} \cdot (1 - \alpha_{1t})/\alpha_{1t}$
  $\boldsymbol{r}_{2t} = (\hat{\boldsymbol{\beta}}_{1t} - \alpha_{1t}\boldsymbol{r}_{1t})/(1 - \alpha_{1t})$

  Denoising of the genetic predictor (see Appendix C)
  $\hat{\boldsymbol{z}}_{1t} = g_z(\boldsymbol{p}_{1t}, \tau_{1t})$
  $\upsilon_{1t} = \langle \tau_1/(\tau_1 + \alpha^2 \boldsymbol{y}^\alpha \cdot e^{-\alpha(\mu + \hat{\boldsymbol{z}}_{1t}) - K}) \rangle$
  $\tau_{2t} = \tau_{1t} \cdot (1 - \upsilon_{1t})/\upsilon_{1t}$
  $\boldsymbol{p}_{2t} = (\hat{\boldsymbol{z}}_{1t} - \upsilon_{1t}\boldsymbol{p}_{1t})/(1 - \upsilon_{1t})$

  EM update of the estimates of Weibull model hyperparameters α and μ (see Appendix E)
  Nonlinear censoring MMSE of the effect sizes (see Appendix D)
  $\hat{\boldsymbol{\beta}}_{2t}^{(0)} = (\tau_{2t}\boldsymbol{X}^T\boldsymbol{X} + \gamma_{2t}\boldsymbol{I})^{-1}(\tau_{2t}\boldsymbol{X}^T\boldsymbol{p}_{2t} + \gamma_{2t}\boldsymbol{r}_{2t})$
  Compute $\hat{\boldsymbol{\beta}}_{2t}$ by solving $F(\boldsymbol{\beta}) = \boldsymbol{0}$ with initialization $\boldsymbol{\beta}_{init} = \hat{\boldsymbol{\beta}}_{2t}^{(0)}$, where
  $F(\boldsymbol{\beta}) = 2(\gamma_{2t}I + \tau_{2t}\boldsymbol{X}^T\boldsymbol{X})\boldsymbol{\beta} + \alpha\boldsymbol{X}_{\mathcal{C}}^T \exp\Big(\alpha(\log(\boldsymbol{y}_{\mathcal{C}}) - \mu - \boldsymbol{X}_{\mathcal{C}}\boldsymbol{\beta}) - K\Big)$
  $\alpha_{2t} = 2\gamma_{2t} \cdot \text{Trace}([J_\beta F(\hat{\boldsymbol{\beta}}_{2t})]^{-1})/P$, where $J_\beta F(\hat{\boldsymbol{\beta}}_{2t}) = 2(\gamma_{2t}\boldsymbol{I} + \tau_{2t}\boldsymbol{X}^T\boldsymbol{X}) + \alpha\nabla^2 g(\hat{\boldsymbol{\beta}}_{2t})$
  $\gamma_{1,t+1} = \gamma_{2t} \cdot (1 - \alpha_{2t})/\alpha_{2t}$
  $\boldsymbol{r}_{1,t+1} = (\hat{\boldsymbol{\beta}}_{2t} - \alpha_{2t}\boldsymbol{r}_{2t})/(1 - \alpha_{2t})$

  MMSE of the latent predictor
  $\hat{\boldsymbol{z}}_{2t} = \boldsymbol{X}\hat{\boldsymbol{\beta}}_{2t}$
  $\upsilon_{2t} = 2 \cdot \tau_{2t} \cdot \text{Trace}(\boldsymbol{X} \cdot [J_\beta F(\hat{\boldsymbol{\beta}}_{2t})]^{-1} \cdot \boldsymbol{X}^T)/N$
  $\tau_{1,t+1} = \tau_{2t} \cdot (1 - \upsilon_{2t})/\upsilon_{2t}$
  $\boldsymbol{p}_{1,t+1} = (\hat{\boldsymbol{z}}_{2t} - \upsilon_{2t}\boldsymbol{p}_{2t})/(1 - \upsilon_{2t})$
end for
return $\hat{\boldsymbol{\beta}}_{1t}$

Handling right censorship with vampW As disease outcomes are only partially observed, with data being frequently right-censored, we revise the LMMSE formulation of the standard VAMP

algorithm for generalized linear models to explicitly incorporate the censored phenotypes. Let $\mathcal{C} = \{i \mid \delta_i = 0\}$ denote the set of indices corresponding to the $N_\mathcal{C}$ censored individuals ($|\mathcal{C}| = N_\mathcal{C}$), $\boldsymbol{y}_\mathcal{C} = [y_{c,1}, \ldots, y_{c,N_\mathcal{C}}] \in \mathbb{R}^{N_\mathcal{C}}$ the censored observations and $\boldsymbol{z}_\mathcal{C} = \boldsymbol{X}_\mathcal{C}\boldsymbol{\beta}$ the corresponding linear predictors, where $\boldsymbol{X}_\mathcal{C}$ represents the matrix containing only information on individuals in $\mathcal{C}$. We note that both the data matrix and the phenotype vector are split into censored and uncensored individuals at runtime, without the need to perform the splits manually beforehand.

We reformulate the problem by incorporating the survival function of the standard Gumbel distribution S_w into the likelihood term of the optimization problem. We recall that the survival function of the standard Gumbel distribution evaluated at $\alpha\left(\log(y_{c,i}) - \mu - (\boldsymbol{X}_\mathcal{C}\boldsymbol{\beta})_i\right) - K$ reflects the probability that the censored individual i survived longer than the censoring time $y_{c,i}$. Motivated by this interpretation, the final optimization task at hand becomes:

$$\hat{\boldsymbol{\beta}}_{2t} = \underset{\boldsymbol{\beta}\in\mathbb{R}^P}{\arg\min} \Bigg\{ \gamma_{2t}\|\boldsymbol{r}_{2t} - \boldsymbol{\beta}\|_2^2 + \tau_{2t}\|\boldsymbol{p}_{2t} - \boldsymbol{X}\boldsymbol{\beta}\|_2^2 - \sum_{i=1}^{N_\mathcal{C}} \log S_w\left(\alpha\left(\log(y_{c,i}) - \mu - (\boldsymbol{X}_\mathcal{C}\boldsymbol{\beta})_i\right) - K\right) \Bigg\}. \tag{7}$$

This objective function represents a Maximum A Posteriori (MAP) estimator for the current VAMP iteration. The quadratic terms incorporate the extrinsic information passed from the VAMP factor graph (approximated as Gaussian priors with precisions γ_{2t} and τ_{2t}), while the logarithmic term explicitly handles the non-Gaussian nature of the censored data by maximizing the survival probability S for the unobserved event times. Following the derivation described in Appendix D, we obtain the following formulae for the Onsager correction terms at iteration t:

$$\alpha_{2t} = \frac{2 \cdot \gamma_{2t} \cdot \text{Trace}([J_{\boldsymbol{\beta}} F(\hat{\boldsymbol{\beta}}_{2t})]^{-1})}{P} \quad \text{and} \quad \upsilon_{2t} = \frac{2 \cdot \tau_{2t} \cdot \text{Trace}(\boldsymbol{X} \cdot [J_{\boldsymbol{\beta}} F(\hat{\boldsymbol{\beta}}_{2t})]^{-1} \cdot \boldsymbol{X}^T)}{N},$$

where $F : \mathbb{R}^P \to \mathbb{R}^P$ is the gradient of the objective function defined as

$$F(\boldsymbol{\beta}) = 2(\gamma_{2t}\boldsymbol{I} + \tau_{2t}\boldsymbol{X}^T\boldsymbol{X})\boldsymbol{\beta} + \alpha\boldsymbol{X}_\mathcal{C}^T \exp\left(\alpha(\log(\boldsymbol{y}_\mathcal{C}) - \mu - \boldsymbol{X}_\mathcal{C}\boldsymbol{\beta}) - K\right).$$

Initialization. The vampW framework allows several parameters to be specified before running the algorithm. Some of these are estimated from the data, while others influence convergence behavior. In our simulations, we intentionally initialize vampW with a mis-specified prior to demonstrate that the algorithm can adapt and learn the prior from the data. Following the prior definition in Equation 4, we initialize the slab part of the prior with unit variance and sparsity $\lambda = 0.3$. While the majority of traits utilize a unit slab part variance, a smaller variance of 0.01 is applied to specific diverged scenarios. In our analysis, we observe better stability of the algorithm when using only 1 slab component ($L = 1$). Furthermore, we initialize the α parameter in the Weibull model to 3 (Equation 3). We fix the damping factor at 0.3 and train the algorithm for 30 iterations. Based on our observations, this number of iterations is sufficient for the hyperparameters to stabilize. Finally, we choose the best iteration based on train concordance index after 5 warm up iterations.

We initialize vampW for all 24 UK Biobank traits with a consistent prior structure, setting the initial shape parameter $\alpha = 3$ and sparsity rate $\lambda = 0.3$ and only one mixture component in the slab part. While the majority of traits utilize a slab variance of 0.01, a wider variance of 1 is applied to specific phenotypes such as hypertension and death. The global shrinkage parameter ρ is initialized between 0.3 and 1.0 depending on the trait, with an adaptive ρ strategy enabled for a subset of runs, including depression, Parkinson's disease, and several cancer endpoints, to allow for dynamic adjustment of update speed. More details can be found in Tables 4-5 in Appendix A. As in the simulations, we choose the final iteration based on the best training concordance index after 5 warm-up iterations.

Association testing. To evaluate the statistical relevance of the markers and test their associations with the traits under investigation, vampW provides Bayesian Posterior Inclusion Probabilities (PIP). Following Algorithm 1, PIPs are calculated using the noisy signal estimates $\boldsymbol{r}_{1,t}$ and the associated precision $\gamma_{1,t}$ at iteration t, relying on the interpretation of those quantities within AMP frameworks

in the standard setting with no censoring. We employ the PIP formulation in (Depope et al., 2026), i.e.,

$$\zeta_j := \frac{\lambda_t \cdot \sum_{i=1}^{L} \pi_{i,t} \cdot \mathcal{N}\big((\boldsymbol{r}_{1,t})_j; 0, \sigma_{i,t}^2 + \gamma_{1,t}^{-1}\big)}{\lambda_t \cdot \sum_{i=1}^{L} \pi_{i,t} \cdot \mathcal{N}\big((\boldsymbol{r}_{1,t})_j; 0, \sigma_{i,t}^2 + \gamma_{1,t}^{-1}\big) + (1-\lambda_t) \cdot \mathcal{N}\big((\boldsymbol{r}_{1,t})_j; 0, \gamma_{1,t}^{-1}\big)}, \tag{8}$$

where ζ_j is the posterior probability that marker j has a non-zero effect, and $\{\lambda, (\pi_i)_{i=1}^L, (\sigma_i^2)_{i=1}^L\}$ are prior parameters (Equation 4) adaptively learned throughout the vampW iterations. The PIP can be viewed as the Bayesian analogue of a p-value. We use a PIP threshold of 0.95, which represents strong evidence that a variable is associated with the outcome and corresponds to a 5% probability of false discovery in a calibrated setting.

Computational efficiency. vampW can be run as a single-threaded program on an HPC cluster, requiring approximately 64 GB of random-access memory (RAM) per trait and an average of 12.7 CPU hours (measured on AMD EPYC 9654 Processor) to complete a single trait analysis. This corresponds to approximately 7.3 GBP when run on a cloud-computing DNAnexus virtual machine (mem3_ssd3_x8)[2], which is required due to data protection. The CoxPH LASSO approach can be parallelized, requiring a total of 160 GB of RAM when using 10 parallel CPU cores. With an average of 60.8 CPU hours required per trait, this corresponds to 10.5 GBP per trait when run on a mem3_ssd3_x24 machine. In comparison, CoxPH requires an average of 0.3 CPU hours per trait, and marginal CoxPH requires 0.1 CPU hours per trait, however those methods score lowest in our benchmarks in out-of-sample prediction and variable selection as we show in Section 4.

Benchmarking in data-scarce regime We conduct a comparison of vampW with deep learning survival methods on several commonly used datasets, namely METABRIC, GBSG, SUPPORT, and SAC3, obtained from the *pycox python* library (Håvard Kvamme, 2024). Following the study (Monod et al., 2025), we examine vampW in the data-scarce regime by subsampling the datasets to 100 samples for training and 25 samples for testing. We recreate the exact subsets used in (Monod et al., 2025), enabling a direct comparison of our results with those reported in the central analysis of the study. We report results for several recent and more traditional deep learning methods, namely: Multi-Task Logistic Regression (MTLR) (Yu et al., 2011), DeepHit (Lee et al., 2018), DeepSurv (Katzman et al., 2018), LogisticHazard (Gensheimer & Narasimhan, 2019), CoxTime (Kvamme et al., 2019), CoxCC (Kvamme et al., 2019), PMF (Kvamme & Borgan, 2019), PCHazard (Kvamme & Borgan, 2019), BCESurv (Kvamme & Borgan, 2023), DySurv (Mesinovic et al., 2024), SumoNet (Rindt et al., 2022), DQS (Yanagisawa, 2023), and NeuralSurv (Monod et al., 2025). Here, we initialize the vampW prior with one slab component of unit variance and 50% probability. We set the initial parameter α to 3 and the damping factor to 0.3, allowing for automatic damping tuning over a maximum of 4 iterations.

4 RESULTS

4.1 SIMULATION STUDY IN UKB PPP

We begin by benchmarking vampW against several state-of-the-art variable selection approaches within a semi-synthetic simulation study. Using the 2,924 observed protein levels from the UKB PPP dataset measured in 53,018 individuals, we simulate artificial phenotypic outcomes in the 24 scenarios described in Section 2.3. We report the false discovery rate (FDR) and true positive rate (TPR) in 4 representative scenarios in Figure 1, where FDR $=$ FP$/$(FP$+$TP) and TPR $=$ TP$/$(TP$+$FN), with FP denoting number of false discoveries, TP the number of true positive discoveries, and FN the number of false negative discoveries. FDR–TPR plots for all 24 simulation scenarios are reported in Figure 5 in Appendix B.

The results of Figure 1 show that joint variable selection under the Weibull model of our vampW approach achieves a calibrated false discovery rate, while outperforming both marginal and joint CoxPH models in terms of true positive rate. More precisely, marginal one-at-a-time association testing using the Cox model produces over 80% false positives across all simulated scenarios, even with a Bonferroni correction. This is due to the fact that, when testing each protein individually,

[2] https://documentation.dnanexus.com/developer/api/running-analyses/instance-types

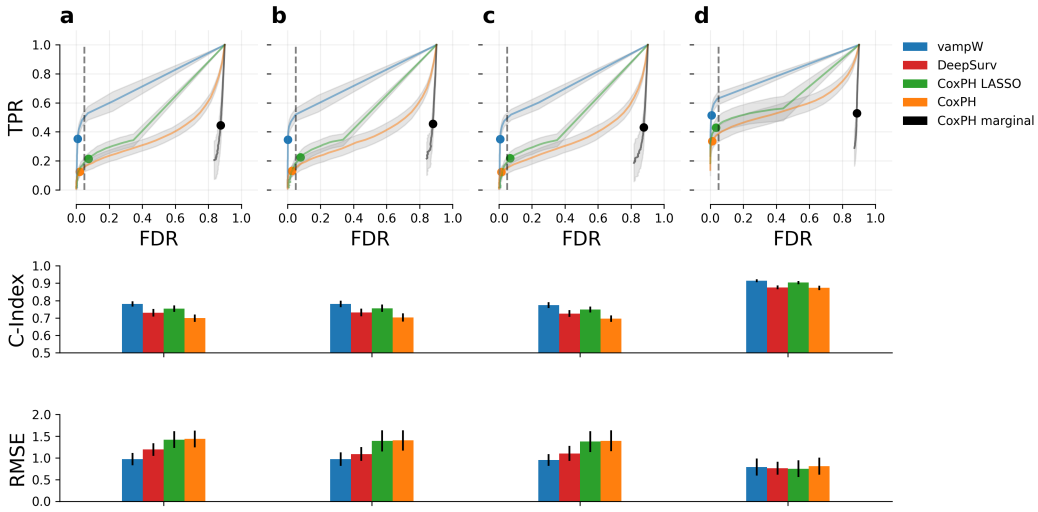

Figure 1: **Simulation study of variable selection and out-of-sample prediction performance using protein data from the UK Biobank.** We present four representative simulation scenarios (out of the 24 in total presented in Figures 5-8 in Appendix B), covering low signal-to-noise ratios, model mis-specification, and high censoring rates. Specifically, subplots (a, b, c, d) correspond to simulation scenarios (m, o, q, t), as described in Table 3. Across all subplots, we use the following color convention: vampW in blue, marginal CoxPH in black, joint CoxPH in orange, LASSO-penalized CoxPH in green, and DeepSurv in red. In the first row, we show variable selection performance as the relationship between false discovery rate (FDR $=$ FP$/$(FP $+$ TP)) and true positive rate (TPR $=$ TP$/$(TP $+$ FN)). For vampW, we use posterior inclusion probabilities (PIP) detailed in Section 3, while for the other methods, p-value testing across multiple thresholds is employed, detailed in Section 2.4. Colored dots indicate the 0.95 threshold for vampW PIP and the Bonferroni-corrected 0.05 threshold for the other methods. The second row presents prediction performance measured by the concordance index (C-index) on a held-out test set. The last row presents prediction performance measured by the Root Mean Squared Error (RMSE) between the true and predicted survival times in the logarithmic domain, on uncensored individuals. The error bars (black lines in the bar plots and gray shaded areas in FDR-TPR plots) represent the standard deviation across 50 simulation replicates.

correlations can inflate false discoveries as the estimated effect is confounded by other correlated proteins. Modeling the data using a single joint CoxPH model substantially reduces false positives, leading to well-calibrated association testing with a false discovery rate below 0.05%. Furthermore, employing a LASSO-penalized CoxPH model with 5-fold cross-validation and stability selection increases power over the joint CoxPH model, while maintaining a well-calibrated false discovery rate in most scenarios. However, the TPR achieved by the LASSO-penalized CoxPH model with stability selection is still lower than that of vampW across all settings. The complete results for all the 24 simulation scenarios in Figure 5 are vastly similar. Overall, the TPR of vampW is higher, with comparable or improved FDR to CoxPH approaches. This holds even under model mis-specification (generalized Gamma distribution with shape parameter $\kappa \in \{0.8, 1.2\}$) or when the prior distribution of simulated regression coefficients follows a spike-and-slab prior with slab component being a Laplace distribution which has heavier tails than the Gaussian slab prior used by vampW.

We investigate the ability of vampW to recover signals from highly correlated proteins, by annotating proteins based on their correlation structure (the top 100 correlated proteins are shown in Figure 6 in Appendix B). If the correlation of a particular protein with any other protein is greater than 0.5 in terms of the absolute value of the Pearson correlation coefficient, it is categorized as a highly correlated protein, with remaining proteins categorized as low-correlated. Figure 7 in Appendix B shows that, among true-positive predictions with PIP ≥ 0.95, the fraction of highly versus low-correlated proteins closely matches the corresponding fraction among ground-truth in the simulation. Thus, vampW has equivalent detection power across proteins of different correlation structure.

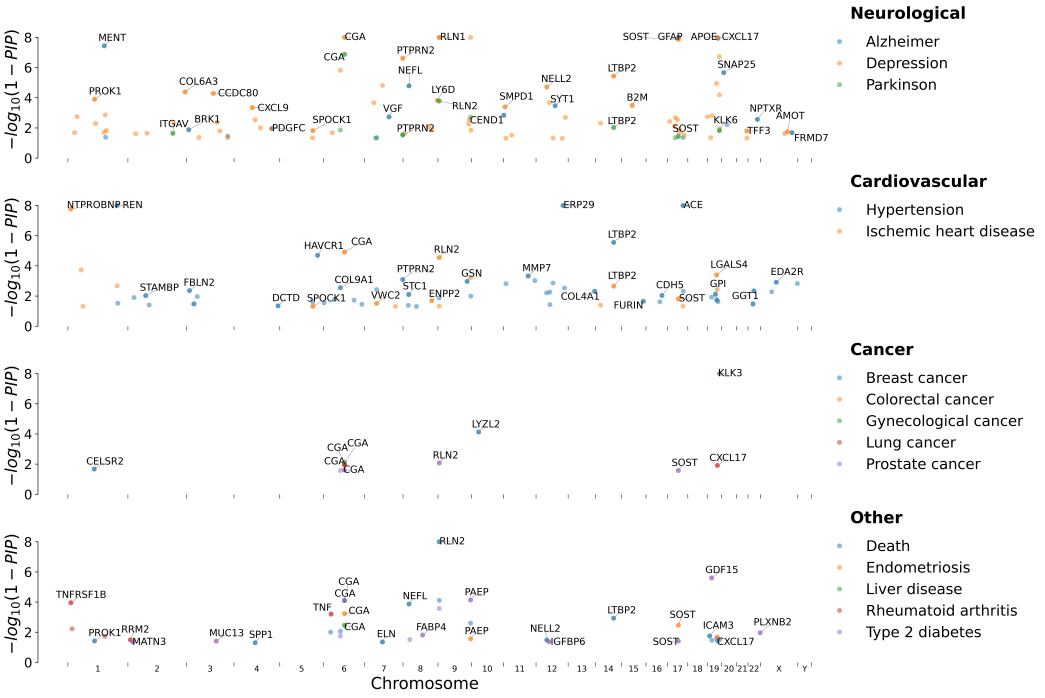

Figure 2: **Protein-coding genes discovered by vampW and their genomic positions.** We show proteins identified by vampW with a posterior inclusion probability (PIP) greater than 0.95. For each chromosome, the top finding by PIP is labeled with the corresponding protein name. Disease outcomes are categorized into four groups: neurological diseases, cardiovascular diseases, cancer types, and other disease-related outcomes.

We then compare the out-of-sample prediction accuracy across approaches, using two metrics: *(i)* Harrell's concordance index (C-index), which measures the proportion of concordant pairs, defined as C-index $= (n_c + 0.5 \cdot n_t)/n_a$, where n_c is the number of pairs which are in the same order in predictions in comparison to the observation, n_t is the number of tied pairs, and n_a is all possible admissible pairs; and *(ii)* root mean squared error (RMSE), which represents the average error between the predicted times and the actual diagnosis times for uncensored individuals, defined as $\text{RMSE}(\boldsymbol{y}, \hat{\boldsymbol{y}}) = (N^{-1} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2)^{1/2}$, where $\hat{\boldsymbol{y}}$ is the predicted outcome time and $\boldsymbol{y}$ is the ground-truth for the uncensored individuals. For the simulation results (Figures 1 and 8), the RMSE is computed on $\log(\boldsymbol{y})$, which is the quantity standardized to zero mean and unit variance by all methods. For the results on real data (Figure 3, discussed in Section 4.2), the RMSE is computed on $\boldsymbol{y}$, which represents the actual time of onset and hence provides a more interpretable quantity.

Prediction accuracies for different methods are presented in Figure 1 for 4 representative scenarios and in Figure 8 (in Appendix B) for all 24 simulation scenarios. These plots show that, in comparison to CoxPH LASSO, vampW achieves higher C-index point estimates across all simulation scenarios, as well as lower RMSE in most scenarios. We additionally compare vampW to DeepSurv, a deep learning extension of the CoxPH model developed for risk prediction. DeepSurv performs better than the standard CoxPH in terms of both metrics, but is outperformed by vampW in all simulation scenarios in terms of C-index. We refrain from comparing association testing performance with DeepSurv, as the latter was primarily developed for prediction tasks (where it does not improve upon vampW) and it does not provide variable selection within the current framework.

4.2 Analyzing 24 disease-related outcomes in UK Biobank

We apply vampW to the age-at-diagnosis of 24 disease outcomes derived from medical records in the UK Biobank, linked to protein measurements from blood samples. In Figure 2, we show vampW-

discovered proteins that pass a PIP threshold of 0.95. For 9 of the 24 traits, no association passes the PIP threshold. We also run marginal CoxPH, joint CoxPH, and LASSO-penalized CoxPH analyses, all using Bonferroni-corrected p-value testing. Marginal testing identifies tens of times more associations compared with joint vampW testing, consistently across most traits (exceptions being Alzheimer's disease, vascular dementia, brain/CNS disorders, and prostate disease). However, our simulation study indicates inflated false discoveries when using marginal testing, which often selects features that appear significant individually but are correlated with each other, whereas joint testing produces a more biologically meaningful set of discoveries, reducing the need for unnecessary follow-up experiments. The set of vampW discoveries is largely independent of the top-x marginal discoveries, where x corresponds to the number of vampW discoveries. Specifically, the overlap between vampW and the top-x marginal CoxPH results is 37 out of 219 vampW associations. This indicates that vampW identifies largely novel associations, and that its joint testing capability cannot be replicated simply by tightening the marginal testing p-value threshold. In comparison with joint CoxPH and LASSO-penalized CoxPH, 61 and 110 associations, respectively, overlap with vampW. The overlap indicates that vampW is not overly conservative in variable selection, as it identifies proteins that are also detected by LASSO.

One of the leading associations we identify is the protein-coding gene *CGA*, which is associated with multiple diseases, including endometriosis, T2D, depression, ischemic heart disease, liver disease, Parkinson's disease, gynecological cancer, prostate cancer, and colorectal cancer. Notably, plasma levels of *CGA* are highly influenced by age and sex in non-linear ways and rank among the top six protein-coding genes (alongside *FSHB*, *SOST*, *GDF15*, *MLN*, and *PTN*) exhibiting the most significant age-dependent changes (Deng et al., 2025; Sigurdsson et al., 2025). This suggests that plasma levels of these proteins may be associated with these outcomes simply because both the proteins and the time-to-diagnosis both vary exponentially with age.

To explore this further, we conduct an additional covariate adjustment including exponential age effects along the linear ones, as described in Section 2.2. We re-analyze four representative traits in Figure 9 (Appendix B), initializing vampW with the same priors as in the main analysis. This yields associations of plasma protein levels with disease onset beyond age-related variation. The protein-coding gene *CGA*, previously reported as age-associated, disappears from ischemic heart disease, colorectal disease, and mortality associations when non-linear effects of age on protein levels are taken into account. Moreover, for certain outcomes, the signal becomes attributed to a different set of proteins, revealing new underlying biological structure. For example, we find the chemokine *CXCL17* as associated with colorectal cancer, which is previously reported as a potential biomarker for the diagnosis of colon cancer (Ohlsson et al., 2016).

Variation in the *APOE* protein, especially the alleles ε_3 and ε_4, has been widely established as a risk factor for late-onset disease across multiple large cohorts (Williams et al., 2026). Regardless of adjustment for non-linear age effects, we find *APOE* to be significantly associated with Alzheimer's disease progression. Additionally, we find *ACE* to be associated with both hypertension and ischemic heart disease, supporting a role in blood pressure regulation (Zambrano et al., 2023). Prostate-specific antigen (PSA) is the most widely used clinical biomarker for identifying men at risk of prostate cancer prior to symptom onset, and its mechanisms have been extensively studied (Pellegrino et al., 2021). PSA is encoded by the *KLK3* gene, which exhibits the strongest association with prostate cancer among all cancer associations in our study. Growth differentiation factor 15 (*GDF15*) reduces food intake and represents a promising therapeutic target for the treatment of type 2 diabetes (Aguilar-Recarte et al., 2022). It is also a useful biomarker for identifying individuals at increased risk of diabetes (Aguilar-Recarte et al., 2022), and vampW exhibits a strong association between GDF15 protein levels and type 2 diabetes. We also highlight the discovery of collagen alpha-1(IX) *COL9A1* in type 2 diabetes, which was also identified by the standard joint CoxPH model but not by CoxPH LASSO. Although it has previously been reported among top plasma proteins (Deng et al., 2025), it is not well established as a biomarker and may provide novel insights into the biological pathways underlying the disease. Furthermore, we identify *SOST* as a novel candidate associated with endometriosis, a potential biomarker that has not been widely studied in this context.

In Figure 3, we show the absolute values of C-index and RMSE metrics across traits with more than 100 uncensored individuals present in the test set. In terms of RMSE, vampW outperforms all other methods across all traits: on average, it reduces the RMSE by 33% with respect to CoxPH, by 32% with respect to CoxPH LASSO and by 26% with respect to DeepSurv. Figure 3 gives the RMSE in

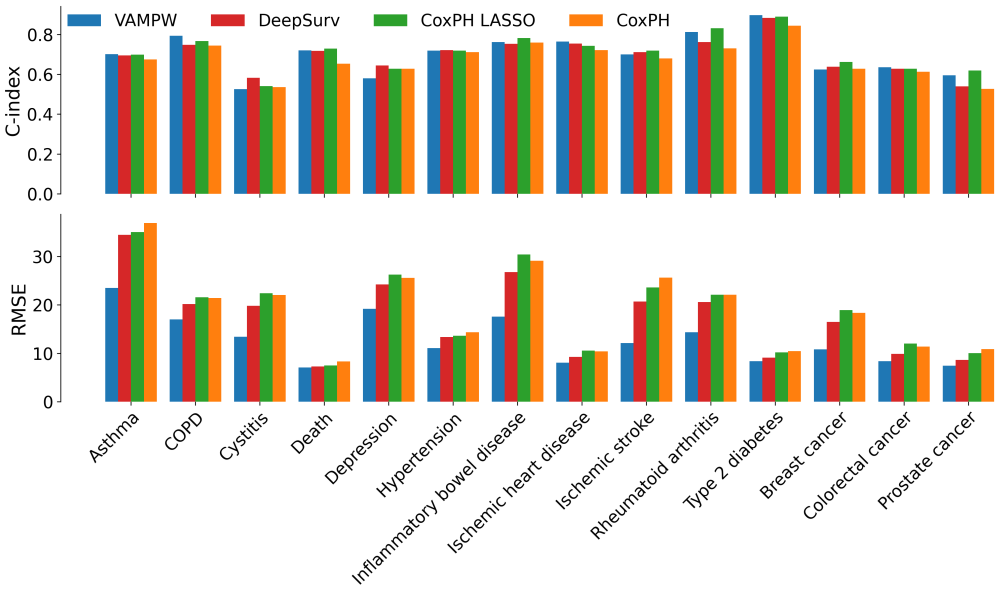

Figure 3: **Out-of-sample performance comparison for CoxPH, LASSO-penalized CoxPH, DeepSurv, and vampW.** We evaluate time-to-diagnosis predictions for 14 health-related outcomes in the UK Biobank (traits with $\geq$ 100 cases in the test set) using protein measurements adjusted for age and sex. Model accuracy is assessed using the Concordance Index (C-index), which measures the probability of correctly ranking patient risks, and the root mean squared error (RMSE), representing the average error in years between predicted and actual diagnosis times for uncensored individuals. Training is performed on data that is first log-transformed and standardized to zero mean and unit variance, and then scaled back using an exponential transformation. To represent actual survival times, we rescale the predictions back to years using statistics from the training set. A sensitivity analysis shows that using test-set statistics for this rescaling impacts results negligibly and preserves the ordering of methods in terms of their performance. In terms of RMSE, vampW significantly improves over all other methods. In terms of C-index, vampW outperforms CoxPH, and it performs on par with CoxPH LASSO and DeepSurv.

the actual scale of years, and we also present the RMSE in the logarithmic time domain, assuming standardized $\log(\boldsymbol{y})$, in Figure 10 (Appendix B), which shows that the relative performance ranking of the methods remains unchanged. In terms of out-of-sample C-index performance, vampW shows an improvement over CoxPH in $11/14$ traits, over LASSO-penalized CoxPH in $6/14$ traits and over DeepSurv in $9/14$ traits. Furthermore, on average, the C-index of vampW is 0.5% better than that of DeepSurv, 4% better than that of CoxPH, and 1.3% worse than that of LASSO-penalized CoxPH. Finally, we note that the largest deviations from the Weibull distribution, as shown in Figure 4 (Appendix B), occur for asthma and inflammatory bowel disease, for which vampW has limited power for association testing but still demonstrates improved time-prediction performance over the various baselines.

To demonstrate the wider applicability of our algorithm outside of the proteomics domain, we additionally benchmark vampW in data-scarce regimes using common survival benchmarking datasets, following the study (Monod et al., 2025). As shown in Figure 11 (Appendix B), vampW outperforms all deep learning methods on the METABRIC, SUPPORT, and SAC3 datasets, and outperforms 6 out of 13 benchmarking methods on the GBSG dataset.

5 Discussion and Concluding Remarks

In this work, we develop a novel algorithm, dubbed vampW, for Bayesian variable selection under the Weibull model. Applying our approach to the largest proteomics data available to date, we demonstrate its capability to jointly fit thousands of plasma proteomics markers across tens of thou-

sands of individuals from the UK Biobank, revealing new disease biomarkers. Our simulation study shows that vampW outperforms state-of-the-art methods for variable selection, reducing false discoveries compared to marginal testing and improving the true positive rate relative to multiple forms of CoxPH models. When analyzing real health-related outcomes from the UK Biobank, vampW identifies 219 associations between protein levels and human complex diseases. For out-of-sample prediction of the outcome ranking (C-index), vampW outperforms the standard CoxPH model and achieves performance comparable to penalized versions of CoxPH and to the deep learning-based method DeepSurv. For the prediction of disease onset times in uncensored individuals, vampW yields more accurate predictions, measured by RMSE, as compared to all other benchmarked methods. We also show that vampW achieves strong out-of-sample prediction performance in other non-proteomics data-scarce regimes, outperforming deep learning methods on most of the examined survival benchmarking datasets, even when trained on very small sample size.

There are several potential avenues for future work. First, plasma protein levels, as well as other multi-omics markers, may be influenced by multiple confounding factors, such as medication use, sex, and age, including non-linear effects of age on protein levels, as suggested by our ablation study. In this context, a key direction for future work is to leverage the medical records of individuals in the UK Biobank, which would allow adjustments for a broader set of potential confounders. Moreover, to examine whether variation in protein levels reflects disease predisposition or the disease state itself, one possible approach is to restrict the analysis to individuals who experience events after the date of blood sampling and assessment, rather than incorporating both prospective and historical events. With more samples available covering a larger portion of the population prevalent with a certain disease, studying associations between survival time and protein levels during the follow-up period represents a viable approach. Finally, our future work will also focus on improving the computational efficiency of the vampW algorithm through parallelization and reduced memory requirements, which has the potential to enable large-scale analyses involving millions of omics markers.

In summary, vampW is a novel approach for selecting multiomics markers associated with human diseases and for predicting disease onset times. With increasing sample sizes and expanding data coverage in the world's largest biobanks, joint vampW analyses have the potential to advance our understanding of the genomic architecture of complex human diseases by enabling interpretable identification of disease associations and providing accurate predictions of disease onset.

ACKNOWLEDGMENTS

We thank members of the Medical Genomics group at ISTA for their comments, which improved this manuscript. We would like to acknowledge the participants and investigators of the UK Biobank study. High-performance computing was supported by the Scientific Service Units (SSU) of IST Austria through resources provided by Scientific Computing (SciComp). MM and AD are funded by the European Union (ERC, INF2, project number 101161364). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council Executive Agency. Neither the European Union nor the granting authority can be held responsible for them.

AUTHOR COMPETING INTERESTS

This study received research funding from Boehringer Ingelheim through a research collaboration agreement with MRR at the Institute of Science and Technology Austria. The remaining authors declare no competing interests.

DATA AVAILABILITY

This project uses the UK Biobank data under project number 35520. UK Biobank proteomics and phenotypic data is available through a formal request at (`http://www.ukbiobank.ac.uk`).

CODE AVAILABILITY

Source code for vampW can be found at `https://github.com/medical-genomics-group/Time2EVAMP`.

REFERENCES

David Aguilar-Recarte, Emma Barroso, Xavier Palomer, Walter Wahli, and Manuel Vázquez-Carrera. Knocking on GDF15's door for the treatment of type 2 diabetes mellitus. *Trends in Endocrinology & Metabolism*, 33(11):741–754, 2022.

M. Austin Argentieri, Sihao Xiao, Derrick Bennett, Laura Winchester, Alejo J. Nevado-Holgado, Upamanyu Ghose, Ashwag Albukhari, Pang Yao, Mohsen Mazidi, Jun Lv, Iona Millwood, Hannah Fry, Rodosthenis S. Rodosthenous, Jukka Partanen, Zhili Zheng, Mitja Kurki, Mark J. Daly, Aarno Palotie, Cassandra J. Adams, Liming Li, Robert Clarke, Najaf Amin, Zhengming Chen, and Cornelia M. Van Duijn. Proteomic aging clock predicts mortality and risk of common age-related diseases in diverse populations. *Nature Medicine*, 30(9):2450–2460, 2024.

Jean Barbier, Florent Krzakala, Nicolas Macris, Léo Miolane, and Lenka Zdeborová. Optimal errors and phase transitions in high-dimensional generalized linear models. *Proceedings of the National Academy of Sciences*, 116(12):5451–5460, 2019.

Jean Barbier, Francesco Camilli, Marco Mondelli, and Manuel Sáenz. Fundamental limits in structured principal component analysis and how to reach them. *Proceedings of the National Academy of Sciences*, 120(30):e2302028120, 2023.

Mohsen Bayati and Andrea Montanari. The dynamics of message passing on dense graphs, with applications to compressed sensing. *IEEE Transactions on Information Theory*, 57(2):764–785, 2011.

Erwin Bolthausen. An iterative construction of solutions of the TAP equations for the Sherrington–Kirkpatrick model. *Communications in Mathematical Physics*, 325(1):333–366, 2014.

Cameron Davidson-Pilon. lifelines: survival analysis in python. *Journal of Open Source Software*, 4(40):1317, 2019.

Yue-Ting Deng, Jia You, Yu He, Yi Zhang, Hai-Yun Li, Xin-Rui Wu, Ji-Yun Cheng, Yu Guo, Zi-Wen Long, Yi-Lin Chen, Ze-Yu Li, Liu Yang, Ya-Ru Zhang, Shi-Dong Chen, Yi-Jun Ge, Yu-Yuan Huang, Le-Ming Shi, Qiang Dong, Ying Mao, Jian-Feng Feng, Wei Cheng, and Jin-Tai Yu. Atlas of the plasma proteome in health and disease in 53,026 adults. *Cell*, 188(1):253–271.e7, 2025.

Al Depope, Jakub Bajzik, Marco Mondelli, and Matthew R. Robinson. Joint modelling of whole genome sequence data for human height via approximate message passing. *Cell Genomics*, 2026. In press.

David L. Donoho, Arian Maleki, and Andrea Montanari. Message Passing Algorithms for Compressed Sensing. *Proceedings of the National Academy of Sciences*, 106:18914–18919, 2009.

Weiwei Duan, Ruyang Zhang, Yang Zhao, Sipeng Shen, Yongyue Wei, Feng Chen, and David C. Christiani. Bayesian variable selection for parametric survival model with applications to cancer omics data. *Human Genomics*, 12(1):49, 2018.

Shaohua Fan, Renzhe Xu, Qian Dong, Yue He, Cheng Chang, and Peng Cui. Stable cox regression for survival analysis under distribution shifts. *Nature Machine Intelligence*, 6(12):1525–1541, 2024.

Oliver Y Feng, Ramji Venkataramanan, Cynthia Rush, Richard J Samworth, et al. A unifying tutorial on approximate message passing. *Foundations and Trends® in Machine Learning*, 15(4): 335–536, 2022.

Alyson K. Fletcher and Philip Schniter. Learning and free energies for vector approximate message passing. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4247–4251, 2017.

Danni A. Gadd, Robert F. Hillary, Zhana Kuncheva, Tasos Mangelis, Yipeng Cheng, Manju Dissanayake, Romi Admanit, Jake Gagnon, Tinchi Lin, Kyle L. Ferber, Heiko Runz, Biogen Biobank Team, Kyle L. Ferber, Christopher N. Foley, Riccardo E. Marioni, and Benjamin B. Sun. Blood protein assessment of leading incident diseases and mortality in the UK biobank. *Nature Aging*, 4(7):939–948, 2024.

Michael F. Gensheimer and Balasubramanian Narasimhan. A scalable discrete-time survival model for neural networks. *PeerJ*, 7:e6257, 2019.

Jessica Gong, Dylan M Williams, Shaun Scholes, Sarah Assaad, Feifei Bu, Shabina Hayat, Paola Zaninotto, and Andrew Steptoe. Unraveling the role of proteins in dementia: insights from two uk cohorts with causal evidence. *Brain Communications*, 7(2):fcaf097, 03 2025.

Tingyang Hu, Qiang Liu, Qile Dai, Aron S. Buchman, David A. Bennett, Shinya Tasaki, Yanling Wang, Nicholas T. Seyfried, Philip L. De Jager, Michael P. Epstein, and Jingjing Yang. Proteome-wide association studies using summary pQTL data of brain, CSF, and plasma identify 30 risk genes of alzheimer's disease dementia. *Alzheimer's Research & Therapy*, 17(1):135, 2025.

M.F. Hutchinson. A stochastic estimator of the trace of the influence matrix for laplacian smoothing splines. *Communications in Statistics - Simulation and Computation*, 19(2):433–450, 1990.

Håvard Kvamme. pycox: Survival analysis with PyTorch, 2024. URL `https://pypi.org/project/pycox`.

Jared L. Katzman, Uri Shaham, Alexander Cloninger, Jonathan Bates, Tingting Jiang, and Yuval Kluger. DeepSurv: personalized treatment recommender system using a cox proportional hazards deep neural network. *BMC Medical Research Methodology*, 18(1):24, 2018.

Wolfgang Kopp. Aging and "age-related" diseases - what is the relation? *Aging Dis.*, 16(3):1316–1346, June 2024.

Chia-Ling Kuo, Peiran Liu, Gabin Drouard, Eero Vuoksimaa, Jaakko Kaprio, Miina Ollikainen, Zhiduo Chen, Luke C. Pilling, Janice L. Atkins, Richard H. Fortinsky, George A. Kuchel, and Breno S. Diniz. A proteomic signature of healthspan. *Proceedings of the National Academy of Sciences*, 122(23):e2414086122, 2025.

Håvard Kvamme and Ørnulf Borgan. Continuous and discrete-time survival prediction with neural networks. *Lifetime Data Analysis*, 27:710 – 736, 2019.

Håvard Kvamme and Ørnulf Borgan. The brier score under administrative censoring: Problems and a solution. *Journal of Machine Learning Research*, 24(2):1–26, 2023.

Håvard Kvamme, Ørnulf Borgan, and Ida Scheel. Time-to-event prediction with neural networks and cox regression. *ArXiv*, abs/1907.00825, 2019.

Changhee Lee, William Zame, Jinsung Yoon, and Mihaela van der Schaar. Deephit: A deep learning approach to survival analysis with competing risks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1), Apr. 2018.

Nicolai Meinshausen and Peter Bühlmann. Stability selection. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 72(4):417–473, 2010.

Munib Mesinovic, Peter Watkinson, and Tingting Zhu. Dysurv: dynamic deep learning model for survival analysis with conditional variational inference. *Journal of the American Medical Informatics Association*, 33(1):112–122, 11 2024.

Marco Mondelli and Ramji Venkataramanan. Approximate message passing with spectral initialization for generalized linear models. In *International Conference on Artificial Intelligence and Statistics*, pp. 397–405. PMLR, 2021.

Mélodie Monod, Alessandro Micheli, and Samir Bhatt. NeuralSurv: Deep survival analysis with bayesian uncertainty quantification, 2025.

Andrea Montanari and Ramji Venkataramanan. Estimation of low-rank matrices via approximate message passing. *Annals of Statistics*, 45(1):321–345, 2021.

Lina Ohlsson, Marie-Louise Hammarström, Gudrun Lindmark, Sten Hammarström, and Basel Sitohy. Ectopic expression of the chemokine CXCL17 in colon cancer cells. *British Journal of Cancer*, 114(6):697–703, 2016.

Sven E Ojavee, Athanasios Kousathanas, Daniel Trejo Banos, Etienne J Orliac, Marion Patxot, Kristi Läll, Reedik Mägi, Krista Fischer, Zoltan Kutalik, and Matthew R Robinson. Genomic architecture and prediction of censored time-to-event phenotypes with a bayesian genome-wide analysis. *Nat. Commun.*, 12(1):2337, April 2021.

Francesco Pellegrino, Arianna Coghi, Giovanni Lavorgna, Walter Cazzaniga, Edoardo Guazzoni, Irene Locatelli, Isabella Villa, Simona Bolamperti, Nadia Finocchio, Massimo Alfano, Roberta Lucianò, Alberto Briganti, Francesco Montorsi, Andrea Salonia, and Ilaria Cavarretta. A mechanistic insight into the anti-metastatic role of the prostate specific antigen. *Translational Oncology*, 14(11):101211, 2021.

Deborah Plana, Geoffrey Fell, Brian M. Alexander, Adam C. Palmer, and Peter K. Sorger. Cancer patient survival can be parametrized to improve trial precision and reveal time-dependent therapeutic effects. *Nature Communications*, 13(1):873, 2022.

Sebastian Pölsterl. scikit-survival: A library for time-to-event analysis built on top of scikit-learn. *Journal of Machine Learning Research*, 21(212):1–6, 2020.

Sundeep Rangan, Philip Schniter, and Alyson K. Fletcher. Vector approximate message passing. *IEEE Transactions on Information Theory*, 65(10):6664–6684, 2019.

David Rindt, Robert Hu, David Steinsaltz, and Dino Sejdinovic. Survival regression with proper scoring rules and monotonic neural networks. In Gustau Camps-Valls, Francisco J. R. Ruiz, and Isabel Valera (eds.), *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pp. 1190–1205, Valencia, Spain, 28–30 Mar 2022. PMLR.

Philip Schniter, Sundeep Rangan, and Alyson K Fletcher. Vector approximate message passing for the generalized linear model. In *2016 50th Asilomar conference on signals, systems and computers*, pp. 1525–1529. IEEE, 2016.

Arnor I. Sigurdsson, Justus F. Gräf, Zhiyu Yang, Kirstine Ravn, Jonas Meisner, Roman Thielemann, Henry Webel, Roelof A. J. Smit, Lili Niu, Matthias Mann, Bjarni Vilhjalmsson, Benjamin M. Neale, Jens-Christian Holm, Andrea Ganna, Torben Hansen, Ruth J. F. Loos, Simon Rasmussen, and FinnGen. Complex genetic effects linked to plasma protein abundance in the UK Biobank. *Nature Communications*, 17(1):533, dec 2025.

Benjamin B. Sun, Joshua Chiou, Matthew Traylor, Christian Benner, Yi-Hsiang Hsu, Tom G. Richardson, Praveen Surendran, Anubha Mahajan, Chloe Robins, Steven G. Vasquez-Grinnell, Liping Hou, Erika M. Kvikstad, Oliver S. Burren, Jonathan Davitte, Kyle L. Ferber, Christopher E. Gillies, Asa K. Hedman, Sile Hu, Tinchi Lin, Rajesh Mikkilineni, Rion K. Pendergrass, Corran Pickering, Bram Prins, Denis Baird, Chia-Yen Chen, Lucas D. Ward, Aimee M. Deaton, Samantha Welsh, Carissa M. Willis, Nick Lehner, Matthias Arnold, Maria A. Wörheide, Karsten Suhre, Gabi Kastenmüller, Anurag Sethi, Madeleine Cule, Anil Raj, Alnylam Human Genetics, AstraZeneca Genomics Initiative, Biogen Biobank Team, Bristol Myers Squibb, Genentech Human Genetics, GlaxoSmithKline Genomic Sciences, Pfizer Integrative Biology, Population Analytics of Janssen Data Sciences, Regeneron Genetics Center, Hyun Ming Kang, Lucy Burkitt-Gray, Eugene Melamud, Mary Helen Black, Eric B. Fauman, Joanna M. M. Howson, Hyun Min Kang, Mark I. McCarthy, Paul Nioi, Slavé Petrovski, Robert A. Scott, Erin N. Smith, Sándor Szalma, Dawn M. Waterworth, Lyndon J. Mitnaul, Joseph D. Szustakowski, Bradford W. Gibson, Melissa R. Miller, and Christopher D. Whelan. Plasma proteomic associations with genetics and health in the UK biobank. *Nature*, 622(7982):329–338, 2023.

Robert Tibshirani. The LASSO method for variable selection in the Cox model. *Statistics in Medicine*, 16(4):385–395, 1997.

Jeremy Vila, Philip Schniter, Sundeep Rangan, Florent Krzakala, and Lenka Zdeborova. Adaptive damping and mean removal for the generalized approximate message passing algorithm. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2021–2025, 2015.

Jia-Hao Wang, Shan-Shan Dong, Wei Huang, Hao-An Wang, Shao-Shan Liu, Xiaoyi Ma, Ren-Jie Zhu, Wei Shi, Hao Wu, Ke Yu, Tian-Pei Zhang, Cong-Ru Wang, Yan Guo, Hanzhong Xue, and Tie-Lin Yang. Blood plasma proteome-wide association study implicates novel proteins in the pathogenesis of multiple cardiovascular diseases. *Cardiovascular Diabetology*, 24(1):312, 2025.

Dylan M. Williams, Sami Heikkinen, Mikko Hiltunen, FinnGen, Neil M. Davies, and Emma L. Anderson. The proportion of alzheimer's disease attributable to apolipoprotein e. *npj Dementia*, 2(1):1, 2026.

Hiroki Yanagisawa. Proper scoring rules for survival analysis. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 39165–39182, Honolulu, Hawaii, USA, 23–29 Jul 2023. PMLR.

Chun-Nam Yu, Russell Greiner, Hsiu-Chin Lin, and Vickie E. Baracos. Learning patient-specific cancer survival distributions as a sequence of dependent regressors. In *Neural Information Processing Systems*, 2011.

Ana Karina Zambrano, Santiago Cadena-Ullauri, Patricia Guevara-Ramírez, Viviana A. Ruiz-Pozo, Rafael Tamayo-Trujillo, Elius Paz-Cruz, Adriana Alexandra Ibarra-Rodríguez, and Nieves Doménech. Genetic diet interactions of ACE: the increased hypertension predisposition in the latin american population. *Frontiers in Nutrition*, 10:1241017, 2023.

A SUPPLEMENTARY TABLES

Table 1: **Description of traits under investigation from the UK Biobank health records.** For cancer traits, cases are defined by the presence of disease-specific ICD-10 codes (field 40006) or ICD-9 codes (field 40013) across all instances. For the non-cancer traits, we analyze time-to-event derived from the first reported occurrence of each disease (category 1712). The column 'Cases' reports the total number of individuals who experienced the event (non-censored observations) within the study cohort.

ABBREVIATION	PHENOTYPE	UK BIOBANK CODE	CASES
CNS	Brain/CNS cancer	100092 (cancer register)	115
MS	Multiple sclerosis	131042	508
MD	Major depression	130894 OR 130896	7,128
SLE	Systemic lupus erythematosus	131894	579
ENDO	Endometriosis	132122	1,031
VD	Vascular dementia	130838	297
GC	Gynecological cancer	100092 (cancer register)	664
ALS	Amyotrophic lateral sclerosis	42028	508
IBD	Inflammatory bowel disease	131626 OR 131628	1,014
LC	Lung cancer	100092 (cancer register)	598
LD	Liver disease	131658	268
AD	Alzheimer's dementia	130836	546
CD	Colorectal cancer	100092 (cancer register)	988
CYS	Cystitis	132054	2,169
RA	Rheumatoid arthritis	131850	1,556
PD	Parkinson's disease	131022	963
IS	Ischemic stroke	131368	1,255
BC	Breast cancer	100092 (cancer register)	1,983
PC	Prostate cancer	100092 (cancer register)	1,671
COPD	COPD	131492	3,394
T2D	Type 2 diabetes	130708 (T1D 130706 absent)	4,760
IHD	Ischemic heart disease	131306	6,438
DEATH	Death	100093 (death register)	5,765
HTN	Hypertension	131286	22,408

Table 2: **ICD-9 and ICD-10 codes used to define cancer phenotypes in the UK Biobank.** Cases were identified by matching these codes against the UK Biobank cancer registry records (ICD-10 field 40006 and ICD-9 field 40013) to determine the age at diagnosis.

PHENOTYPE	ICD-10 CODE	ICD-9 CODE
Breast Cancer	C500, C501, C502, C503, C504, C505, C506, C508, C509	1740, 1741, 1742, 1743, 1744, 1745, 1746, 1748, 1749, 175
Prostate Cancer	C61	185
Colorectal Cancer	C180, C181, C182, C183, C184, C185, C186, C187, C188, C189, C19, C20, C210, C211, C212, C218	1530, 1531, 1532, 1533, 1534, 1535, 1536, 1537, 1538, 1539, 1540, 1541, 1542, 1543, 1548
Lung Cancer	C33, C340, C341, C342, C343, C348, C349	1620, 1622, 1623, 1624, 1625, 1628, 1629
Gynecological Cancer	C51, C510, C518, C519, C52, C530, C531, C539, C540, C541, C542, C543, C549, C55, C56, C570, C571, C574, C577, C578, C579	179, 1800, 1801, 1808, 1809, 181, 1820, 1828, 1830, 1832, 1840, 1841, 1844, 1848, 1849
Brain/CNS Cancer	C700, C709, C710, C711, C712, C713, C714, C715, C716, C717, C718, C719, C720, C722, C723, C724, C725, C729	1910, 1911, 1912, 1914, 1916, 1917, 1918, 1919, 1920, 1921, 1922

Table 3: **Description of parameters varied in the simulation study.** We simulate 24 artificial phenotypic outcomes covering a broad range of parameterizations, varying the variance explained by protein levels (σ_G), the censoring ratio (C), the phenotype distribution, the ExpGamma parameter (κ), and the distribution of regression coefficients ($\boldsymbol{\beta}$).

LABEL	σ_G^2	C	**PHENOTYPE DISTRIBUTION**	κ	**PRIOR β DISTRIBUTION**
a	40%	90%	Weibull	-	Normal
b	80%	90%	Weibull	-	Normal
c	40%	90%	ExpGamma	0.8	Normal
d	80%	90%	ExpGamma	0.8	Normal
e	40%	90%	ExpGamma	1.2	Normal
f	80%	90%	ExpGamma	1.2	Normal
g	40%	90%	Weibull	-	Laplace
h	80%	90%	Weibull	-	Laplace
i	40%	90%	ExpGamma	0.8	Laplace
j	80%	90%	ExpGamma	0.8	Laplace
k	40%	90%	ExpGamma	1.2	Laplace
l	80%	90%	ExpGamma	1.2	Laplace
m	40%	95%	Weibull	-	Normal
n	80%	95%	Weibull	-	Normal
o	40%	95%	ExpGamma	0.8	Normal
p	80%	95%	ExpGamma	0.8	Normal
q	40%	95%	ExpGamma	1.2	Normal
r	80%	95%	ExpGamma	1.2	Normal
s	40%	95%	Weibull	-	Laplace
t	80%	95%	Weibull	-	Laplace
u	40%	95%	ExpGamma	0.8	Laplace
v	80%	95%	ExpGamma	0.8	Laplace
w	40%	95%	ExpGamma	1.2	Laplace
x	80%	95%	ExpGamma	1.2	Laplace

Table 4: **Prior initialization configurations for vampW.** For analyzing real UK Biobank traits, we use several vampW initializations, varying the damping factor (ρ), the initial Weibull shape parameter (α), and the prior variances and probabilities.

LABEL	ρ	**INITIAL** α	**VARIANCES**	**PROBABILITIES**	**ADAPTIVE** ρ
a	0.3	3	0, 0.01	0.7, 0.3	No
b	0.5	3	0, 0.01	0.7, 0.3	No
c	0.5	3	0, 0.01	0.7, 0.3	Yes
d	0.5	3	0, 1	0.7, 0.3	Yes
e	0.6	3	0, 1	0.7, 0.3	No
f	0.8	3	0, 0.01	0.7, 0.3	No
g	0.8	3	0, 0.01	0.7, 0.3	Yes
h	1.0	3	0, 1	0.7, 0.3	No

Table 5: **Initialization settings used per UK Biobank trait.** We use different prior initializations, described in Table 4, across the various traits. The 'Initialization Label' links each specific disease outcome to the corresponding hyperparameter configuration defined in Table 4, ensuring reproducibility of the analysis settings.

TRAIT	INITIALIZATION LABEL
Alzheimer	b
Asthma	b
COPD	g
Cystitis	g
Death	h
Depression	c
Endometriosis	e
Hypertension	d
Inflammatory Bowel Disease	a
Ischemic Heart Disease	f
Ischemic Stroke	g
Liver Disease	g
Multiple Sclerosis	f
Parkinson	c
Rheumatoid Arthritis	g
Schizophrenia	g
Systemic Lupus Erythematosus	f
Type 2 Diabetes (T2D)	b
Vascular Dementia	f
Breast cancer	g
Brain CNS cancer	g
Colorectal cancer	g
Gynecological cancer	g
Lung cancer	g
Prostate cancer	g

B SUPPLEMENTARY FIGURES

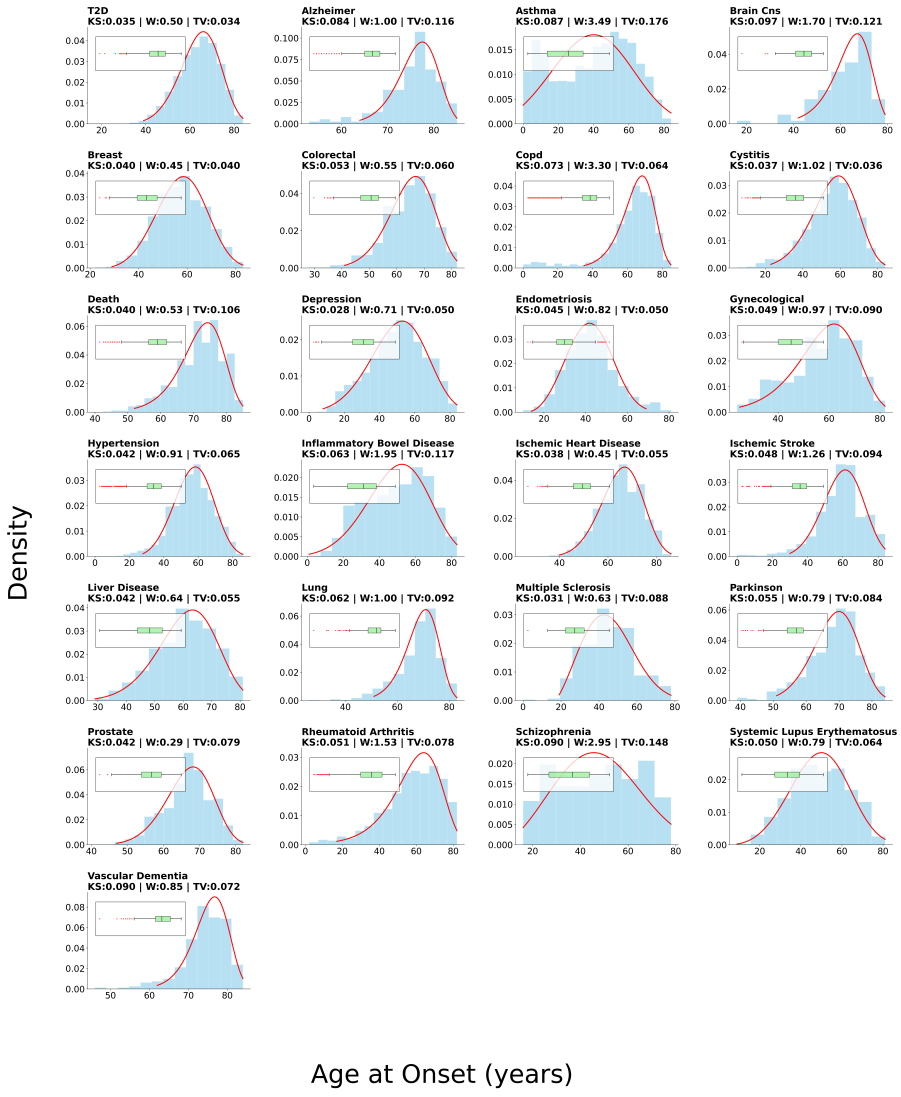

Figure 4: **Analysis of age-at-onset distributions for UK Biobank Traits**. The histograms display the empirical distribution of age-at-onset for cases across various phenotypes in the training set. The blue bars represent the density of observed onset ages, while the solid red line indicates the fitted Weibull probability density function (PDF). Outliers were excluded from the fitting process using the Interquartile Range (IQR) method. The title of each subplot provides goodness-of-fit metrics, including the Kolmogorov-Smirnov statistic (KS), Wasserstein distance (W), and Total Variation distance (TV).

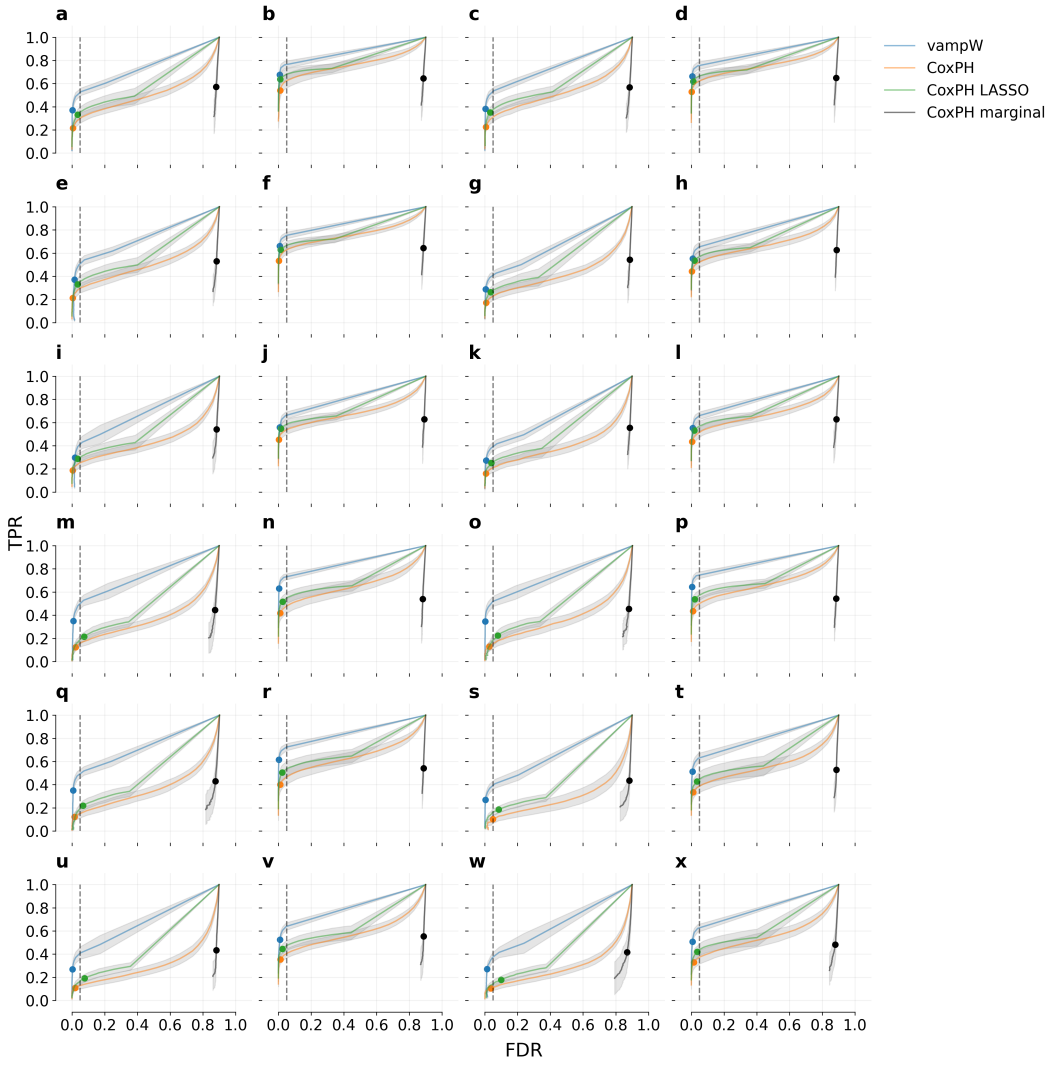

Figure 5: **Simulation study of variable selection performance using protein data from the UK Biobank.** We simulate artificial phenotypic outcomes under 24 scenarios, corresponding to the parameterizations described in Table 3, varying the proportion of variance explained, level of censoring, phenotype distribution, and distribution of regression coefficients. We compare the variable selection performance of vampW against several benchmarking methods (CoxPH, CoxPH LASSO and CoxPH marginal) in terms of the relationship between the false discovery rate (FDR) and true positive rate (TPR). For vampW, we use posterior inclusion probabilities (PIPs), with colored dots indicating a 0.95 threshold. For the other methods, p-value testing across multiple thresholds is employed, with colored dots indicating a Bonferroni-corrected 0.05 threshold. vampW maintains a calibrated FDR and consistently outperforms CoxPH-based models in TPR across all simulation scenarios, including those involving model misspecification. The error bars (gray shaded areas) represent the standard deviation across 50 simulation replicates.

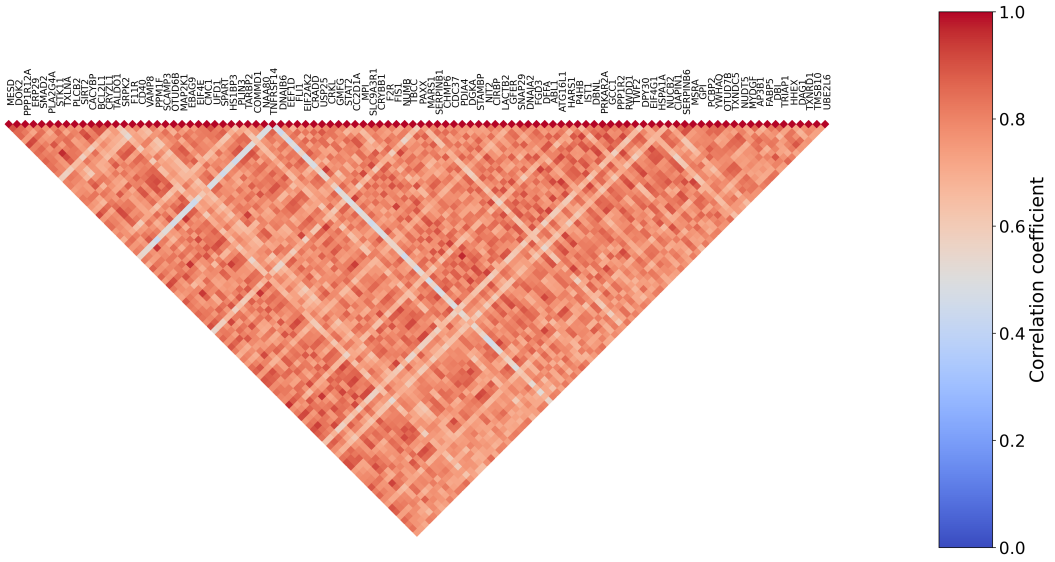

Figure 6: **Correlation structure of the protein measures in the UK Biobank.** We calculate Pearson product-moment correlation coefficients for all proteins, not-adjusted for covariates, and select the top 100 most correlated ones. We plot the correlation structure along with corresponding protein-coding genes.

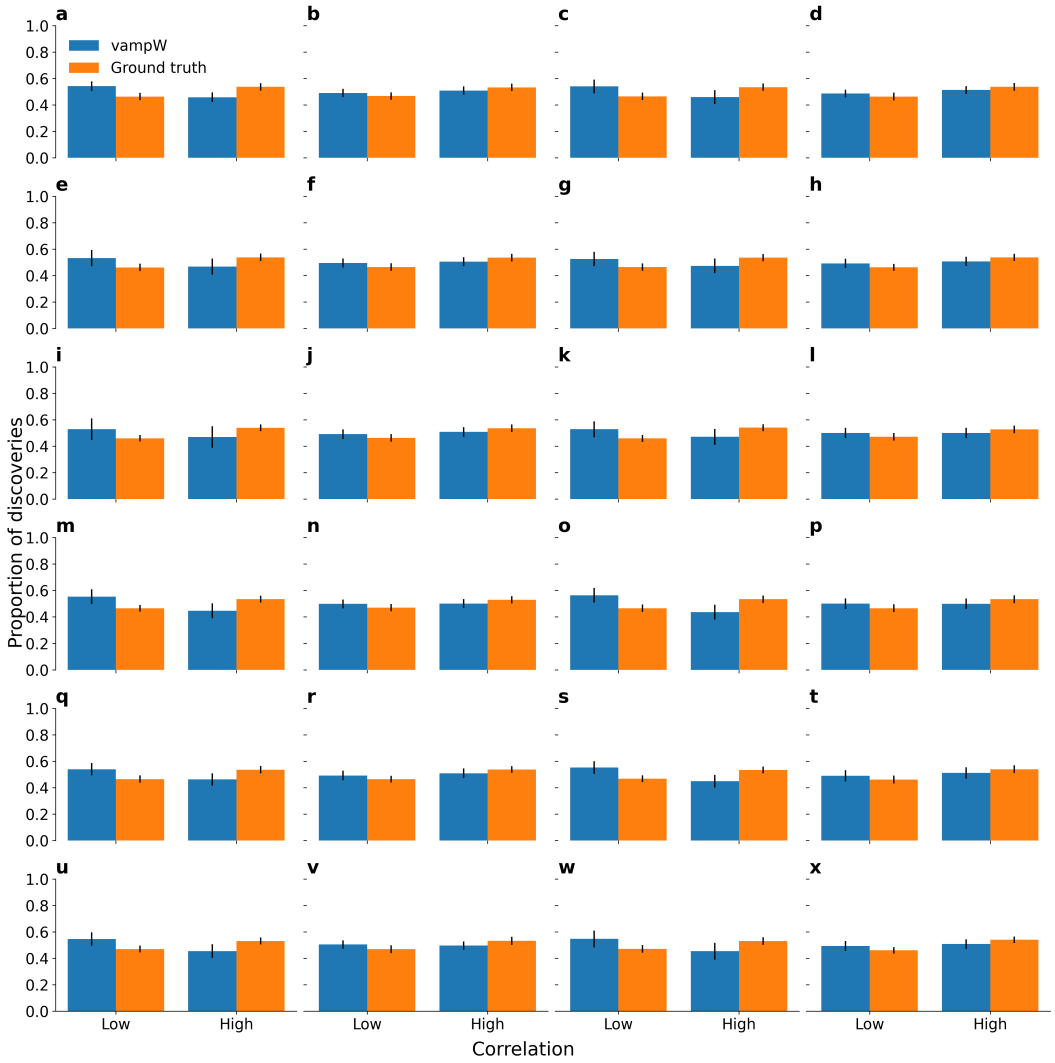

Figure 7: **Protein annotation based on correlation structure**. For simulation scenarios, described in Table 3, we visualize the relative proportion of highly and low-correlated true positive discoveries with respect to all true positive discoveries (blue), and compare this to the proportion of highly and low-correlated causal proteins with respect to all causal proteins (orange). If the correlation of a particular protein with any other protein is greater than 0.5 (absolute value of Pearson correlation coefficient), it is categorized as a highly correlated protein. The remaining proteins are categorized as low-correlated. The black bars show 1 standard error over 30 simulation replicas. The results suggest that vampW detects causal signals even for proteins with correlations above the 0.5 threshold.

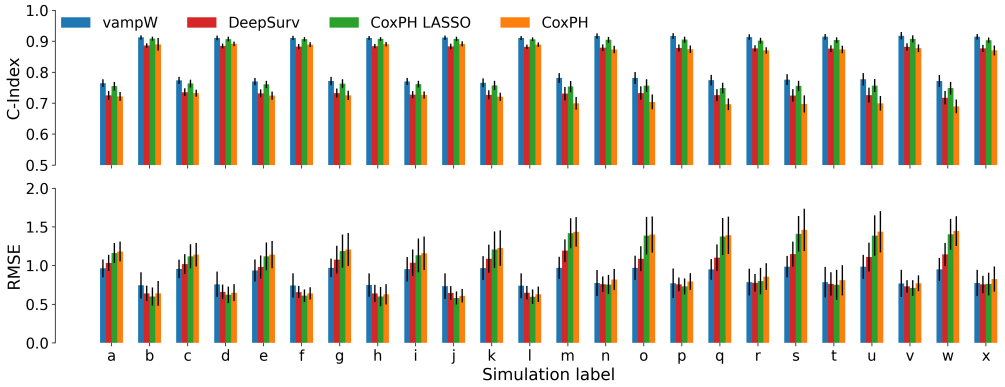

Figure 8: **Simulation study of out-of-sample prediction performance using protein data from the UK Biobank.** We evaluate predictive performance for time of onset across 24 simulated phenotypic outcomes corresponding to the parameterizations in Table 3. We compare vampW with CoxPH, CoxPH LASSO, and DeepSurv in terms of concordance index (C-index) and root mean squared error (RMSE). The RMSE is evaluated in the logarithmic domain, assuming standardized outcomes with zero mean and unit variance. Across all simulated scenarios, VAMPW consistently demonstrated a clear improvement in the C-index when compared to both DeepSurv and the standard CoxPH model. When compared to CoxPH LASSO, the point estimate of C-index values for VAMPW improve, although they remain within overlapping error bars across all scenarios. In terms of RMSE, VAMPW strictly outperformed the standard CoxPH model in six scenarios and improved upon CoxPH LASSO in five scenarios. In all remaining comparisons, the predictive performance of the evaluated methods fell within each other's error margins. The reported error bars represent the standard deviation across 50 simulation replicates (black lines at the top of the colored bars).

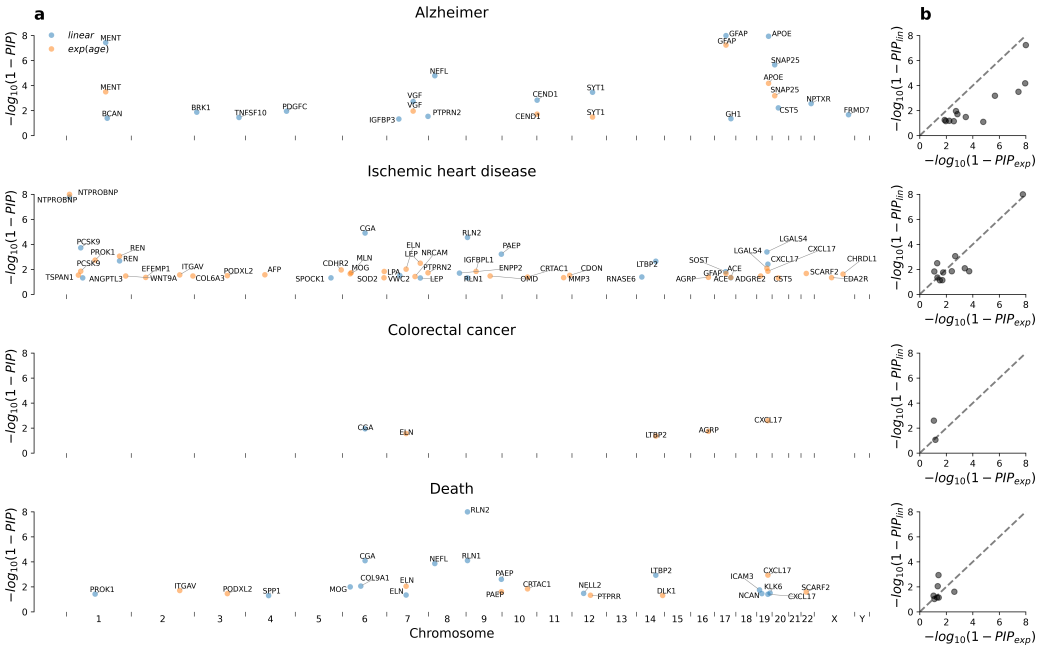

Figure 9: **Adjusting for non-linear effects of age on protein levels**. In addition to the standard linear adjustment (blue) for the age covariate, we evaluate the vampW model using protein data adjusted for exponential age effects (orange). In column (a), we show significant protein-coding genes at their corresponding DNA positions that pass a PIP threshold of 0.95 for four representative outcomes, namely Alzheimer's disease, ischemic heart disease, colorectal cancer, and death. In column (b), we show the relationship between PIPs passing a threshold of 0.9 across the two models, where PIP_{lin} and PIP_{exp} denote the standard linear and exponential age adjustments, respectively. Notably, the gene *CGA*, previously reported as age-associated in recent work, disappears from the discoveries when considering non-linear effects of age on protein levels.

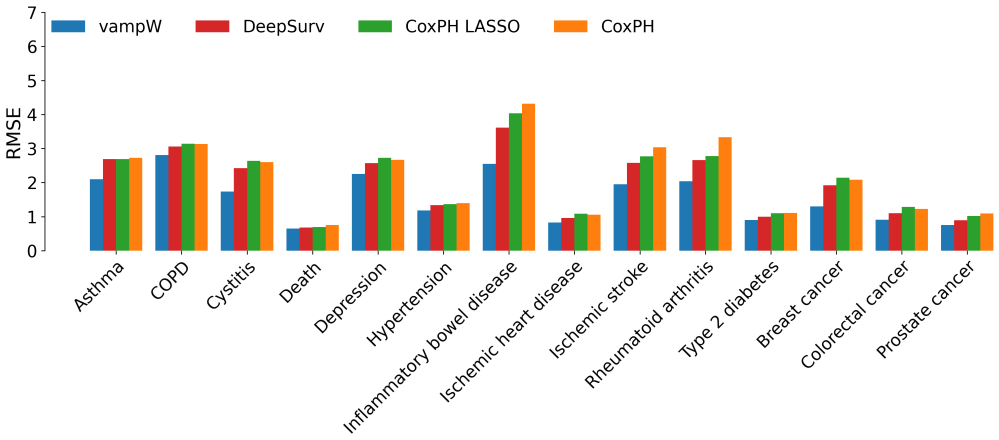

Figure 10: **Out-of-sample performance comparison in terms of RMSE on the log-normalized times for CoxPH, LASSO-penalized CoxPH, DeepSurv, and vampW.** We evaluate time-to-diagnosis predictions for 14 health-related outcomes in the UK Biobank (traits with $\geq$ 100 cases present in the test set) using protein measurements adjusted for age and sex. The accuracy of the model is assessed using the root mean squared error (RMSE). The training is performed on data that is first log-transformed and standardized to zero mean and unit variance, and then scaled back using an exponential transformation. The RMSE calculation presented in this figure is performed on the logarithmic scale in which the data is standardized, and it includes only uncensored individuals across traits.

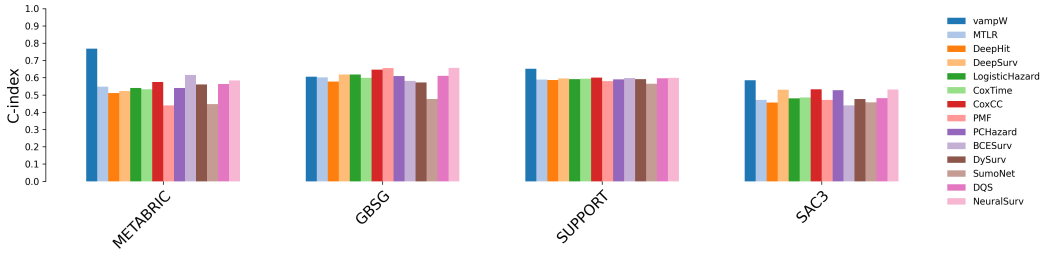

Figure 11: **Out-of-sample performance comparison in terms of the Concordance index for vampW and deep learning methods.** We compare predictions of vampW with common deep learning methods on several widely used survival benchmarking datasets, namely METABRIC, GBSG, SUPPORT, and SAC3. The benchmarking results for deep learning are reported in a recent NeuralSurv study Monod et al. (2025), which addresses prediction in data-scarce regimes. vampW outperforms all deep learning methods on METABRIC, SUPPORT, and SAC3 datasets, and outperforms 6 out of 13 benchmarking methods on the GBSG dataset.

C Details on denoiser for Weibull link function

We assume observations are independent, allowing for component-wise Maximum A Posteriori estimation. Proceeding with the construction of an age-at-onset denoiser for individual i, we express the posterior as follows, excluding terms independent of z_i:

$$\arg\max_{z_i} \left\{ \exp\left(-\alpha z_i - y_i^{\alpha} \cdot \exp(-\alpha(\mu + z_i) - K) \right) \cdot \exp\left(-\frac{(z_i - p_{1,i})^2 \cdot \tau_1}{2} \right) \right\}.$$

Let us define $g_z : \mathbb{R} \to \mathbb{R}$ as

$$g_z(z_i) = -\alpha z_i - y_i^{\alpha} \cdot e^{-\alpha(\mu_i + z_i) - K} - \frac{(z_i - p_{1,i})^2 \cdot \tau_1}{2}.$$

Then, taking the derivative with respect to z_i, one obtains

$$-\alpha + \alpha e^{-\alpha(\mu_i + z_i) - K} \cdot y^{\alpha} - \tau_1(z_i - p_1) = 0,$$

which can be solved using fixed-point iterations. In addition, we note that

$$g_z''(z_i) = -\alpha^2 \cdot y_i^{\alpha} \cdot e^{-\alpha(\mu_i + z_i) - K} - \tau_1 < 0, \quad \text{for all } z_i,$$

and, hence, g is a strictly concave function (see a visualization in Figure 12).

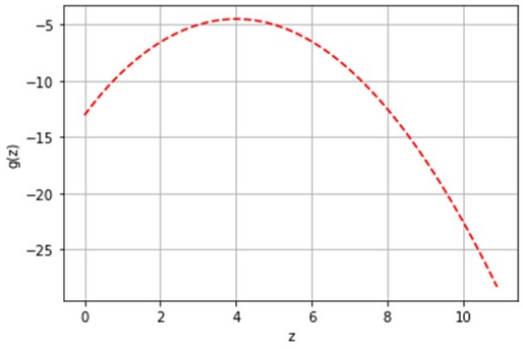

Figure 12: Visualization of a function $g_z(z_i)$ for $\alpha = 1, \tau_1 = 1, p = 5, \mu_i = 0, y_i = 1$

Furthermore, by invoking the Implicit Function Theorem, we can compute the derivative of the denoiser as

$$\frac{dz}{dp_1} = \frac{\tau_1}{\tau_1 + \alpha^2 y_i^{\alpha} \cdot e^{-\alpha(\mu_i + z_i) - K}}.$$

D DETAILS ON NON-LINEAR MMSE HANDLING RIGHT-CENSORSHIP

Lemma 1. *Differentiating the function inside the optimization problem (7) with respect to $\boldsymbol{\beta}$ and equating the result to zero yields*

$$2\gamma_{2t}(\boldsymbol{\beta} - \boldsymbol{r}_{2t}) + 2\tau_{2t}\boldsymbol{X}^T(\boldsymbol{X}\beta - \boldsymbol{p}_{2t}) + \alpha\boldsymbol{X}_{\mathcal{C}}^T h(\alpha(\log(\boldsymbol{y}_{\mathcal{C}}) - \mu - \boldsymbol{X}_{\mathcal{C}}\boldsymbol{\beta}) - K) = 0,$$

where $h : \mathbb{R} \to \mathbb{R}$ is the standard hazard function of the Gumbel model given by $h(x) = e^x$, and the notation $h(\boldsymbol{v})$ with $\boldsymbol{v} \in \mathbb{R}^N$ denotes the component-wise application of the function h. Furthermore, the Onsager corrections for the denoiser and nonlinear MMSE step are equal to

$$\alpha_{2t} = \frac{2 \cdot \gamma_{2t} \cdot \mathit{Trace}([J_\beta F(\hat{\boldsymbol{\beta}}_{2t})]^{-1})}{P} \quad \textit{and} \quad \upsilon_{2t} = \frac{2 \cdot \tau_{2t} \cdot \mathit{Trace}(\boldsymbol{X} \cdot [J_\beta F(\hat{\boldsymbol{\beta}}_{2t})]^{-1} \cdot \boldsymbol{X}^T)}{N}.$$

Proof. Let us define the function $g : \mathbb{R}^P \to \mathbb{R}$ with

$$g(\boldsymbol{\beta}) = -\sum_{i=1}^{N_{\mathcal{C}}} \log S(\alpha(\log(y_{c,i}) - \mu - (\boldsymbol{X}_{\mathcal{C}}\boldsymbol{\beta})_i) - K),$$

where $N_{\mathcal{C}}$ is the number of censored individuals in the study and $\boldsymbol{X}_{\mathcal{C}}$ is the design matrix corresponding to those individuals. By performing differentiation, it follows that

$$\begin{aligned}\frac{\partial g(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}_j} &= \sum_{i=1}^{N_{\mathcal{C}}} \alpha \cdot h(\alpha(\log(y_{c,i}) - \mu - (\boldsymbol{X}_{\mathcal{C}}\boldsymbol{\beta})_i) - K) \cdot (\boldsymbol{X}_{\mathcal{C}})_{i,j} \\ &= \alpha\langle h(\alpha(\log(y_{c,i}) - \mu - (\boldsymbol{X}_{\mathcal{C}}\beta)_i) - K), (\boldsymbol{X}_{\mathcal{C}})_{:,j}\rangle.\end{aligned}$$

Hence,

$$\nabla_{\boldsymbol{\beta}} g(\boldsymbol{\beta}) = \alpha \cdot \boldsymbol{X}_{\mathcal{C}}^T h(\alpha(\log(y_{c,i} - \mu - (\boldsymbol{X}_{\mathcal{C}}\boldsymbol{\beta})_i) - K).$$

Furthermore,

$$\begin{aligned}\frac{\partial g(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}_k \partial \boldsymbol{\beta}_j} &= \frac{\partial}{\partial \boldsymbol{\beta}_k}\left[\sum_{i=1}^{N_{\mathcal{C}}} \alpha \cdot h(\alpha(\log \boldsymbol{y}_{\mathcal{C}} - \mu - \boldsymbol{X}_{\mathcal{C}}\boldsymbol{\beta}) - K) \cdot (\boldsymbol{X}_{\mathcal{C}})_{i,j}\right] \\ &= \alpha \sum_{i=1}^{N_{\mathcal{C}}} h'(\alpha(\log y_{c,i} - \mu - (\boldsymbol{X}_{\mathcal{C}}\boldsymbol{\beta})_i) - K) \cdot (\boldsymbol{X}_{\mathcal{C}})_{i,j} \cdot (\boldsymbol{X}_{\mathcal{C}})_{i,k} \\ &= \alpha\langle h'(\alpha(\log y_{c,i} - \mu - (\boldsymbol{X}_{\mathcal{C}}\boldsymbol{\beta})_i) - K) \cdot (X_{\mathcal{C}})_{:,j} \cdot (\boldsymbol{X}_{\mathcal{C}})_{:,k}\rangle,\end{aligned}$$

with $h'(x) = e^x$. It directly follows that

$$\nabla^2 g(\boldsymbol{\beta}) = \boldsymbol{X}_{\mathcal{C}}^T \text{Diag}(h'(\alpha(\log \boldsymbol{y}_{\mathcal{C}} - \mu - \boldsymbol{X}_{\mathcal{C}}\boldsymbol{\beta}) - K))\boldsymbol{X}_{\mathcal{C}}.$$

We now define $F : \mathbb{R}^P \to \mathbb{R}^P$ as

$$F(\beta) = 2(\gamma_{2t}\boldsymbol{I} + \tau_{2t}\boldsymbol{X}^T\boldsymbol{X})\boldsymbol{\beta} + \alpha\boldsymbol{X}_{\mathcal{C}}^T h(\alpha(\log(\boldsymbol{y}_{\mathcal{C}}) - \mu - \boldsymbol{X}_{\mathcal{C}}\boldsymbol{\beta}) - K),$$

which means that a solution $\hat{\boldsymbol{\beta}}_{2t}$ to the optimization problem (7) satisfies

$$\hat{\boldsymbol{\beta}}_{2t} = F^{-1}(2\gamma_{2t}\boldsymbol{r}_{2t} + 2\tau_{2t}\boldsymbol{X}^T\boldsymbol{p}_{2t}).$$

Note that

$$\begin{aligned}\text{div}_{r_{2t}}(F^{-1}(2\gamma_{2t}r_{2t} + 2\tau_{2t}\boldsymbol{X}^T\boldsymbol{p}_{2t})) &= \text{Trace}(J_\beta F^{-1}(2\gamma_{2t}r_{2t} + 2\tau_{2t}\boldsymbol{X}^T\boldsymbol{p}_{2t})) \cdot 2\gamma_{2t} \\ &= \text{Trace}([J_\beta F(F^{-1}(2\gamma_{2t}r_{2t} + 2\tau_{2t}\boldsymbol{X}^T\boldsymbol{p}_{2t}))]^{-1}) \cdot 2\gamma_{2t} \\ &= \text{Trace}([J_\beta F(\hat{\boldsymbol{\beta}}_{2t})]^{-1}) \cdot 2\gamma_{2t},\end{aligned}$$

where

$$J_\beta F(\hat{\beta}_{2t}) = 2(\gamma_{2t}I + \tau_{2t}\boldsymbol{X}^T\boldsymbol{X}) + \alpha\nabla^2 g(\hat{\boldsymbol{\beta}}_{2t}).$$

Finally, we conclude that

$$\alpha_{2t} = \frac{2 \cdot \gamma_{2t} \cdot \text{Trace}([J_\beta F(\hat{\boldsymbol{\beta}}_{2t})]^{-1})}{P}.$$

Similarly, we derive

$$\begin{aligned}
\text{div}_{p_{2t}}(XF^{-1}(2\gamma_{2t}\boldsymbol{r}_{2t} + 2\tau_{2t}\boldsymbol{X}^T\boldsymbol{p}_{2t})) &= \text{Trace}(\boldsymbol{X}J_\beta F^{-1}(2\gamma_{2t}\boldsymbol{r}_{2t} + 2\tau_{2t}\boldsymbol{X}^T\boldsymbol{p}_{2t})) \\
&= \text{Trace}(\boldsymbol{X}J_\beta F^{-1}(\hat{\boldsymbol{\beta}}_{2t}) \cdot 2\tau_{2t} \cdot \boldsymbol{X}^T) \\
&= 2\tau_{2t} \cdot \text{Trace}(\boldsymbol{X} \cdot [J_\beta F(\hat{\beta}_{2t})]^{-1} \cdot \boldsymbol{X}^T),
\end{aligned}$$

which gives the Onsager correction

$$\upsilon_{2t} = \frac{2 \cdot \tau_{2t} \cdot \text{Trace}(\boldsymbol{X} \cdot [J_\beta F(\hat{\boldsymbol{\beta}}_{2t})]^{-1} \cdot \boldsymbol{X}^T)}{N}.$$

□

E DETAILS ON EM HYPERPARAMETER UPDATES

We present the derivation of the expectation-maximization update formulae for the hyperparameters μ and α under the Weibull model for non-censored phenotypic outcomes. We define the E-step and M-step of the EM algorithm at iteration t of vampW in the following way:

1. E-step: The approximation of the posterior distribution typical for E-steps happens implicitly in VAMP-type algorithms, where the following approximate posterior probability distribution for the genetic component, $\boldsymbol{z} = \boldsymbol{X}\boldsymbol{\beta}$, is obtained:
$$b^{(t)}(\boldsymbol{z}) = \mathcal{N}(\boldsymbol{z}; \hat{\boldsymbol{z}}_{\boldsymbol{2t}}^{(t)}, \boldsymbol{I}_N/\xi^{(t)}).$$
2. M-step: We aim to maximize the evidence likelihood over μ and α
$$(\alpha^{(t+1)}, \mu^{(t+1)}) = \text{argmax}_{\mu,\alpha}\{L^{(t)}(\mu, \alpha)\},$$
where we define
$$\begin{aligned} L^{(t)}(\mu, \alpha) &= \sum_{i=1}^{N} \int b_i^{(t)}(z_i) \cdot \log p(y_i|z_i, \alpha, \mu) dz_i \\ &= \sum_{i=1}^{N} \int \mathcal{N}(z_i; \hat{z}_i^{(t)}, 1/\xi^{(t)}) \cdot \log p(y_i|z_i, \alpha, \mu) dz_i. \end{aligned}$$
Here, $b_i^{(t)}(z_i)$ is the marginal distribution of the i-th component of $\boldsymbol{z}$ obtained from $b^{(t)}(\boldsymbol{z})$ and $\mathcal{N}(x; \mu, \sigma^2)$ denotes the probability density function of a Gaussian with mean μ and variance σ^2 evaluated at x. Furthermore, note that
$$\log p(y_i|z_i, \alpha, \mu) = \log(\alpha) + \alpha(\log y_i - \mu - z_i) - K - e^{\alpha(\log y_i - \mu - z_i) - K}.$$

Update formula for α:

By taking the derivative of $L(\mu, \alpha)$ with respect to α and equating to zero we obtain:
$$\frac{\partial L^{(t)}(\mu, \alpha)}{\partial \alpha} = 0 = \sum_{i=1}^{N} \int \mathcal{N}(z_i; \hat{z}^{(t)}, 1/\xi^{(t)}) \cdot \left\{ \frac{1}{\alpha} + (\log y_i - \mu - z_i) \cdot \left[1 - e^{\alpha(\log y_i - \mu - z_i) - K} \right] \right\} dz_i.$$
Separating the term in brackets results in
$$\begin{aligned} \sum_{i=1}^{N} \left[\frac{1}{\alpha} + \log y_i - \mu - \hat{z}_i^{(t)} \right] &= \frac{N}{\alpha} + \sum_{i=1}^{N} [\log y_i - \mu - \hat{z}_i^{(t)}] \\ &= \sum_{i=1}^{N} \mathcal{N}(z_i; \hat{z}_i^{(t)}, 1/\xi^{(t)}) \cdot (\log y_i - \mu - z_i) \cdot e^{\alpha(\log y_i - \mu - z_i) - K} dz_i \\ &= \sum_{i=1}^{N} \int \mathcal{N}(\log y_i - \mu - \phi_i; \hat{z}_i^{(t)}; 1/\xi^{(t)}) \cdot \phi_i \cdot e^{\alpha\phi_i - K} d\phi_i \\ &= e^{-K} \cdot \sum_{i=1}^{N} \exp(\alpha(\log y_i - \mu - \hat{z}_i^{(t)}) + \frac{\alpha^2}{2\xi^{(t)}}) \cdot \left((\log y_i - \mu - \hat{z}_i^{(t)}) + \frac{\alpha}{\xi^{(t)}} \right). \end{aligned}$$
Finally, we define $f_i(\alpha) := e^{a_i\alpha + b\alpha^2/2} \cdot (a_i + \alpha b)$, where $a_i = \log y_i - \mu - \hat{z_i}^{(t)}$ and $b = \frac{1}{\xi^{(t)}}$. It follows that
$$f'(\alpha) = e^{a\alpha + b\alpha^2/2}(a^2 + b + 2ab\alpha + b^2\alpha^2) > 0 \quad \text{for all } \alpha > 0,$$
i.e. the function is strictly increasing in α. This implies that the first derivative with respect to α of the likelihood function ${L^{(t)}}'$ is
$$\frac{\partial L^{(t)}(\mu, \alpha)}{\partial \alpha} = \text{const} + \frac{N}{\alpha} - e^{-K} \sum_{i=1}^{N} f_i(\alpha).$$

The second derivative is then given by

$$\frac{\partial^2 L^{(t)}(\mu,\alpha)}{\partial^2 \alpha} = -\frac{N}{\alpha^2} - e^{-K}\sum_{i=1}^{N} f_i'(\alpha) < 0, \quad \text{for all } \alpha > 0.$$

We thus have that the second derivative with respect to α is negative on the considered domain and hence the root of $\partial_\alpha L^{(t)}(\alpha,\mu) = 0$ attains a unique global maximum on the considered domain. We solve for the global maximizer α^* using numerical methods (SciPy optimize).

Update formula for μ:

Similarly, by taking the partial derivative of $L^{(t)}(\mu,\alpha)$ with respect to μ and equating it to zero, we obtain

$$\frac{\partial L^{(t)}(\mu,\alpha)}{\partial \mu} = 0 = \sum_{i=1}^{N}\int \mathcal{N}(z_i;\hat{z}_i^{(t)}, 1/\xi^{(t)})\cdot\alpha\cdot[e^{\alpha(\log y_i-\mu-z_i)-K}-1]dz_i,$$

where we use that

$$\frac{\partial \log p(y_i|z_i,\alpha,\mu)}{\partial \mu} = -\alpha + \alpha e^{\alpha(\log y_i-\mu-z_i)-K}.$$

By separating the term in brackets it follows that

$$\begin{aligned}
0 &= \alpha\sum_{i=1}^{n}\int \mathcal{N}\left(z_i;\hat{z}_i^{(t)},\frac{1}{\xi^{(t)}}\right)e^{\alpha(\log y_i-\mu-z_i)-K}dz_i - \alpha\sum_{i=1}^{n}\int \mathcal{N}\left(z_i;\hat{z}_i^{(t)},\frac{1}{\xi^{(t)}}\right)dz_i \\
&= \alpha\sum_{i=1}^{n}\exp\left(-K+\frac{\alpha^2}{2\xi^{(t)}}+\alpha(\log y_i-\hat{z}_i^{(t)}-\mu)\right) \\
&\qquad - \alpha N\exp\left(-K+\frac{\alpha^2}{2\xi^{(t)}}-\alpha\mu\right)\sum_{i=1}^{n}\exp\left(\alpha(\log y_i-\hat{z}_i^{(t)})\right) \\
&= N-K+\frac{\alpha^2}{2\xi^{(t)}}-\alpha\mu+\log\left(\sum_{i=1}^{n}\exp\left(\alpha(\log y_i-\hat{z}_i^{(t)})\right)\right) \\
&= \log N.
\end{aligned}$$

Hence, the update formula for μ satisfies

$$\mu = \frac{-K}{\alpha}+\frac{\alpha}{2\xi^{(t)}}+\frac{1}{\alpha}\log\left(\sum_{i=1}^{n}\exp\left(\alpha(\log y_i-\hat{z}_i^{(t)})\right)\right)-\frac{1}{\alpha}\log N.$$

Furthermore, the maximization problem at hand is concave:

$$\frac{\partial^2 L^{(t)}}{\partial \mu^2} = -\alpha\exp\left(-K+\frac{\alpha^2}{2\xi^{(t)}}-\alpha\mu\right)\sum_{i=1}^{n}\exp\left(\alpha(\log y_i-\hat{z}_i^{(t)})\right)-N<0 \quad \text{for all } \alpha>0$$

Therefore, μ satisfying $\frac{\partial L^{(t)}}{\partial \mu}=0$ attains the global maximum of $L^{(t)}(\mu)$.

F Details on Weibull and ExpGamma outcomes

For the Weibull model, we calculate the scaling factor

$$c = \sqrt{\left(\frac{1}{\sigma_G^2} - 1\right) \cdot \frac{\text{Var}(\boldsymbol{X}\boldsymbol{\beta})}{\sigma_w^2}}, \tag{9}$$

where $\sigma_w^2 = \pi^2/6$ is the variance of Gumbel distribution. Then, we generate time-to-event phenotypes as:

$$\log(\boldsymbol{y}) = \mu + \boldsymbol{X}\boldsymbol{\beta} + c \cdot (\boldsymbol{w} + \kappa). \tag{10}$$

This formulation means that, in the phenotype $\log(y)$, the targeted proportion of variance is explained by markers.

For ExpGamma-distributed phenotypes, we exploit the fact that if $x \sim \text{ExpGamma}(\kappa, \theta, \tilde{\mu})$, then $ax + b \sim \text{ExpGamma}(\kappa, a\theta, a\tilde{\mu} + b)$. Then, we rewrite the model as follows:

$$\log y_i = \mu + \boldsymbol{x}_i^T \boldsymbol{\beta} + \theta(w_i - \psi^{(0)}(\kappa)), \quad w_i \sim \text{ExpGamma}(\kappa, 1, 0),$$

with shape parameter κ, scale parameter θ, and location parameter μ. We note that

$$\mathbb{E}[\log y_i] = \theta\psi^{(0)}(\kappa) + \tilde{\mu},$$
$$\text{Var}[\log y_i] = \theta^2\psi^{(1)}(\kappa),$$

where $\psi^{(m)}(\cdot)$ stands for the polygamma function of order m. Using the parametrization

$$\theta = \sqrt{\left(\frac{1}{\sigma_G^2} - 1\right) \cdot \frac{\text{Var}(\boldsymbol{X}\boldsymbol{\beta})}{\psi^{(1)}(\kappa)}}, \tag{11}$$

we obtain the time-to-event phenotypic outcomes, with the desired proportion of variance explained by the simulated markers.