Appendices for Generalized Linear Multiple Kernel Learning for High-Dimensional Image Data

Qiyuan Shi[a], Jian Kang[a], Yi Li[a,*]

[a]*Department of Biostatistics, University of Michigan, Ann Arbor, MI 48109, USA*

Appendix A. Algorithm

Appendix A.1. GLIMARK Algorithm

For completeness, we restate the key identities used in Algorithm 1—namely the objective function, representer expansion, and selection rule:

$$\begin{aligned} H &= -\frac{1}{N}\sum_{i=1}^{N}\ell(f_i \mid y_i) + \lambda P(\|f\|_{\mathcal{H}_k}) \\ &= -\frac{1}{N}\sum_{i=1}^{N}[y_i f_i - b_{glm}(f_i)] + \lambda P(\|f\|_{\mathcal{H}_k}), \qquad \text{(Appendix A.1)} \\ f_i &= \sum_{j\in S^{(t+1)}}\alpha_j[\boldsymbol{K}]_{i,j} + b, \qquad \text{(Appendix A.2)} \\ j^{*(t+1)} &= \arg\max_{j:j\notin S^{(t)}}\left|\frac{\partial H(\boldsymbol{\alpha}, b)}{\partial \alpha_j}\right|. \qquad \text{(Appendix A.3)} \end{aligned}$$

The proposed GLIMARK is summarized by Algorithm 1. The algorithm supports a variety of response data types. As an example, with $P(\cdot)$ chosen the square function, the objective function H in the algorithm can be

*Corresponding author: Yi Li (email: yili@umich.edu).

Algorithm 1 GLIMARK

Let $b^{(0)} = g_{glm}(p_+)$, $p_+ = \frac{1}{N}\sum_{i=1}^{N} y_i$, $tol = 10^{-6}$
Let $S^{(0)} := \emptyset$
Let $\boldsymbol{\alpha}^{(0)} = \mathbf{0}$
for $t := 0$ to T_{max} **do**
    Select $j^{*(t+1)}$ according to (Appendix A.3)
    if $\left| \frac{\partial H}{\partial \alpha_{j^{*(t+1)}}} \right|_{(\boldsymbol{\alpha}, b) = (\boldsymbol{\alpha}^{(t)}, b^{(t)})} \leq tol$ **then**
        return $\boldsymbol{\alpha}^{(t)}, b^{(t)}$ as the final estimates, i.e., $\hat{\boldsymbol{\alpha}}$ and $\hat{b}$
    end if
    Update $S^{(t+1)}$ as $S^{(t+1)} = S^{(t)} \cup \{j^{*(t+1)}\}$
    Minimize (Appendix A.1) with (Appendix A.2) using Adam, give updates for $\alpha_j^{(t+1)}, \quad \forall j \in S^{(t+1)}$ and $b^{(t+1)}$. Let $\alpha_j^{(t+1)} = 0, \quad \forall j \notin S^{(t+1)}$
end for
return $\boldsymbol{\alpha}^{(t+1)}, b^{(t+1)}$ as the final estimates, i.e., $\hat{\boldsymbol{\alpha}}$ and $\hat{b}$

specified for the Bernoulli and Poisson distributions as:

$$H(\boldsymbol{\alpha}, b) = \begin{cases} -\frac{1}{N}\sum_{i=1}^{N} \left[y_i f_i - \log(1 + \exp(f_i)) \right] + \lambda \|f\|_{\mathcal{H}_k}^2, \\ \text{(Bernoulli)} \\ -\frac{1}{N}\sum_{i=1}^{N} \left[y_i f_i - \exp(f_i) \right] + \lambda \|f\|_{\mathcal{H}_k}^2, \\ \text{(Poisson)} \end{cases}$$

with the gradients given by:

$$\frac{\partial H(\boldsymbol{\alpha}, b)}{\partial \alpha_j} = \begin{cases} -\frac{1}{N}\sum_{i=1}^{N} \left[y_i - \frac{\exp(f_i)}{1+\exp(f_i)} \right] [\boldsymbol{K}]_{i,j} + \lambda \frac{\partial \|f\|_{\mathcal{H}_k}^2}{\partial \alpha_j}, \\ \text{(Bernoulli)} \\ -\frac{1}{N}\sum_{i=1}^{N} \left[y_i - \exp(f_i) \right] [\boldsymbol{K}]_{i,j} + \lambda \frac{\partial \|f\|_{\mathcal{H}_k}^2}{\partial \alpha_j}. \\ \text{(Poisson)} \end{cases}$$

We express the RKHS norm of f as

$$\|f\|_{\mathcal{H}_k}^2 = \boldsymbol{\alpha}^T \tilde{\boldsymbol{K}} \boldsymbol{\alpha},$$

where $\tilde{\boldsymbol{K}}$ is a $J \times J$ block diagonal matrix derived from $\boldsymbol{K}$ (Bi et al., 2004):

$$\tilde{\boldsymbol{K}} = \text{diag}\left[\boldsymbol{K}_{1,1}, \quad \ldots, \quad \boldsymbol{K}_{1,Q}, \quad \boldsymbol{K}_{2,1}, \quad \ldots, \quad \boldsymbol{K}_{2,Q}, \quad \ldots \quad \boldsymbol{K}_{P,1}, \quad \ldots, \quad \boldsymbol{K}_{P,Q} \right].$$

In Step 5 of the algorithm, at iteration t, we define $\tilde{\boldsymbol{\alpha}}^{(t)}$ as the subvector of $\boldsymbol{\alpha}^{(t)}$ containing only the entries indexed by $S^{(t)}$. Correspondingly, we define a block diagonal matrix $\tilde{\mathbf{K}}^{(t)} \in \mathbb{R}^{t\times t}$, which is the submatrix of $\tilde{\mathbf{K}}$ with rows and columns indexed by $S^{(t)}$.

Then when maximizing

$$\left| \frac{\partial H(\boldsymbol{\alpha}, b)}{\partial \alpha_j} \Big|_{(\boldsymbol{\alpha},b)=(\boldsymbol{\alpha}^{(t)},b^{(t)})} \right|$$

we use

$$\| f(\boldsymbol{\alpha}, b)\|^2_{\mathcal{H}_k}\big|_{(\boldsymbol{\alpha},b)=(\boldsymbol{\alpha}^{(t)},b^{(t)})} = \tilde{\boldsymbol{\alpha}}^{(t)T}\tilde{\boldsymbol{K}}^{(t)}\tilde{\boldsymbol{\alpha}}^{(t)}$$

and

$$\frac{\partial \|f(\boldsymbol{\alpha}, b)\|^2_{\mathcal{H}_k}}{\partial \boldsymbol{\alpha}}\Big|_{(\boldsymbol{\alpha},b)=(\boldsymbol{\alpha}^{(t)},b^{(t)})} = 2\tilde{\boldsymbol{K}}^{(t)}\tilde{\boldsymbol{\alpha}}^{(t)}.$$

This formulation enhances computational efficiency by avoiding direct use of $\tilde{\boldsymbol{K}}$, which is large and sparse.

Appendix A.2. Nyström Approximation for GLIMARK

For computational scalability, we approximate each kernel block $\boldsymbol{K}_{p,q} \in \mathbb{R}^{N\times N}$ using the Nyström method. Specifically, for each (p, q), we select a set $\mathcal{L} = \{\ell_1, \ldots, \ell_d\}$ of d landmark indices uniformly without replacement, where $d \ll N$ (Drineas et al., 2005). Define

$$\boldsymbol{C}_{p,q} = \boldsymbol{K}_{p,q}(:, \mathcal{L}) \in \mathbb{R}^{N\times d}, \qquad \boldsymbol{W}_{p,q} = \boldsymbol{K}_{p,q}(\mathcal{L}, \mathcal{L}) \in \mathbb{R}^{d\times d}.$$

We then approximate $\boldsymbol{K}_{p,q}$ by

$$\boldsymbol{K}^{\mathrm{Ny}}_{p,q} = \boldsymbol{C}_{p,q}\boldsymbol{W}^{\dagger}_{p,q}\boldsymbol{C}^{T}_{p,q},$$

where $(\cdot)^{\dagger}$ denotes the Moore–Penrose pseudoinverse.

Accordingly, the block diagonal matrix in the RKHS norm is approximated by

$$\tilde{\boldsymbol{K}}^{\mathrm{Ny}} = \mathrm{diag}\left[\boldsymbol{K}^{\mathrm{Ny}}_{1,1}, \ \ldots, \ \boldsymbol{K}^{\mathrm{Ny}}_{1,Q}, \ \boldsymbol{K}^{\mathrm{Ny}}_{2,1}, \ \ldots, \ \boldsymbol{K}^{\mathrm{Ny}}_{2,Q}, \ \ldots \ \boldsymbol{K}^{\mathrm{Ny}}_{P,1}, \ \ldots, \ \boldsymbol{K}^{\mathrm{Ny}}_{P,Q}\right].$$

Thus, the RKHS norm is approximated by

$$\|f\|^2_{\mathcal{H}_k} \approx \boldsymbol{\alpha}^T\tilde{\boldsymbol{K}}^{\mathrm{Ny}}\boldsymbol{\alpha}.$$

Similarly, at iteration t, we define $\tilde{\boldsymbol{K}}^{(t),\mathrm{Ny}}$ as the submatrix of $\tilde{\boldsymbol{K}}^{\mathrm{Ny}}$ with rows and columns indexed by $S^{(t)}$.

Appendix B. Simulation

Appendix B.1. Simulation Recovery

The first aim of the simulation was to evaluate whether GLIMARK could recover the underlying signal structure, including linear effects, nonlinear effects, and possible interactions. This aim was examined in Scenarios A–E. Scenarios A–C considered linear and nonlinear functional forms, whereas Scenarios D–E considered interaction structures. The emphasis in these scenarios was therefore on recovery and attribution rather than on tuning predictive performance. We assessed whether the active groups or variable combinations received larger estimated contributions than inactive components and whether those contributions were assigned to kernels consistent with the underlying functional form.

For the first aim, we followed a setup similar to (Friedman et al., 2010), with 10 groups of variables, denoted by G1 through G10, each containing 10 variables independently generated from the standard normal distribution. The numbers of active variables in G1 through G6 were 10, 8, 6, 4, 2, and 1, respectively, whereas G7 through G10 contained no active variables. In the following, $\boldsymbol{z}_{i,g}$ represents the active variables in group g, and $\boldsymbol{\beta}_g$ denotes their corresponding coefficients, which were randomly assigned values of $+1$ or -1. Outcomes were binary in all five scenarios and were generated according to

$$p(y \mid \theta) = \exp\left\{y\theta - b_{\text{glm}}(\theta) + c_{\text{glm}}(y)\right\}. \qquad \text{(Appendix B.1)}$$

For each scenario, we conducted 100 independent simulation replicates with a sample size of 400 per replicate. To isolate differences in recovery behavior from differences caused by hyperparameter selection, GLIMARK was fitted using the same prespecified complexity parameters in every replicate, with $\lambda = 0.001$ and $T = 400$. Kernel configurations were selected according to the structure being examined in each scenario, including linear, degree-2 polynomial, RBF, or combined-kernel models.

Recovery was also compared with commonly used benchmark methods, including unregularized logistic regression, logistic regression with ℓ_1 or ℓ_2 regularization, and random forest. The settings used for each benchmark are labeled directly in the corresponding figures.

Component contributions were defined as follows:

1. For GLIMARK, the contribution of component c was based on the

absolute fitted representer coefficients:

$$I_c^{\text{GLIMARK}} = \sum_{j \in \mathcal{J}_c} |\widehat{\alpha}_j| \,,$$

where $\mathcal{J}_c$ denotes the representer terms associated with component c. Here, contributions were computed at the component level by aggregating over all representer terms belonging to that component. In the simulations, each component corresponded directly to one feature group $G_1, \ldots, G_{10}$; thus, the component contribution may equivalently be written as $I_{\text{Group}=G_g}$ for group G_g. The same applies to the other contribution metrics below.

2. For the regression-based benchmarks, the contribution of component c was based on the absolute fitted regression coefficients:

$$I_c^{\text{Reg}} = \sum_{j \in \mathcal{J}_c} \left|\widehat{\beta}_j\right| ,$$

where $\mathcal{J}_c$ denotes the variables or interaction terms associated with component c.

3. For random forest, the contribution of component c was based on aggregated Gini importance:

$$I_c^{\text{RF}} = \sum_{j \in \mathcal{J}_c} \text{GiniImp}_j \,.$$

Within each fitted model, component contributions were converted to percentages of the total contribution:

$$I_c^{(\%)} = 100 \frac{I_c}{\sum_{c'} I_{c'}}.$$

The figures report the mean contribution percentage for each component together with its standard deviation.

Scenario A: $f(\boldsymbol{z}_i) = \sum_{g=1}^{6} \langle \boldsymbol{z}_{i,g}, \boldsymbol{\beta}_g \rangle + 1$**.** A linear GLIMARK correctly identified the active groups, with I_{Group} contributions generally reflecting the number of active variables and decreasing from 19.17% in G1 to 7.37% in G6. Minor contributions of approximately 5% appeared in the inactive groups. Adding an RBF kernel did not change the overall detection pattern: the linear components continued to decrease across G1–G6, while the RBF components

were small and nearly uniform across all groups (3.34%–3.96%), indicating no additional nonlinear structure. Logistic regression and its ℓ_1- and ℓ_2-regularized variants similarly recovered the decreasing contribution pattern across the active groups. Random forest also assigned greater importance to the earlier active groups, but distributed importance more uniformly and showed weaker separation between active and inactive groups.

Scenario B: $f(\boldsymbol{z}_i) = \langle \boldsymbol{z}_{i,1}^3, \boldsymbol{\beta}_1 \rangle + \langle \boldsymbol{z}_{i,2}^2, \boldsymbol{\beta}_2 \rangle + \langle \boldsymbol{z}_{i,3}, \boldsymbol{\beta}_3 \rangle + 1$**.** A linear GLIMARK failed to capture the quadratic effect in G2, assigning it a contribution of 7.14%, similar to those of the inactive groups. Adding a degree-2 polynomial kernel improved detection: the G2 polynomial component contributed 9.75%, increasing its total group contribution to 12.02%. This suggests that the polynomial kernel captured relevant nonlinear structure that was not distinguished by the linear model. The logistic regression models similarly emphasized G1 and G3 but assigned G2 contributions comparable to those of the inactive groups. Random forest assigned somewhat greater importance to G2 than to most inactive groups, but the separation was less distinct than the kernel-specific detection provided by GLIMARK. Among the evaluated methods, the multiple-kernel GLIMARK model was the only one to clearly distinguish the G2 effect from the inactive groups, highlighting its ability to detect nonlinear structure through kernel-specific attribution.

Scenario C: $f(\boldsymbol{z}_i) = 3\sum_{g=1}^{6} \langle \boldsymbol{z}_{i,g}^{\exp}, \boldsymbol{\beta}_g \rangle + \langle \boldsymbol{z}_{i,4}, \boldsymbol{\beta}_4 \rangle + 1$**.** A linear GLIMARK detected the linear effect in G4, assigning it a contribution of 17.51%, but failed to distinguish the exponential contributions in G1–G6, with the remaining groups receiving contributions of approximately 9%. The RBF GLIMARK successfully identified the exponential effects in G1 and G2, with contributions of 51.68% and 44.47%, respectively, but did not retain the linear effect in G4. In the combined linear and RBF model, the contributions were distributed across kernels, with the linear component continuing to emphasize G4 while the RBF components assigned larger contributions to G1 and G2. Logistic regression and its regularized variants primarily detected the linear effect in G4 and did not distinguish the exponential groups from inactive groups. Random forest also assigned its largest contribution to G4, while showing somewhat elevated contributions for G1–G3 but weaker separation than the standalone RBF GLIMARK. These results highlight the sensitivity of estimated contributions to kernel choice and the relative magnitudes of the underlying effects.

Scenario D: $f(\boldsymbol{z}_i) = \langle \boldsymbol{z}_{i,1}, \boldsymbol{\beta}_1 \rangle + \langle \boldsymbol{z}_{i,2}, \boldsymbol{\beta}_2 \rangle + \sum_{j<k} \gamma_{jk} z_{i,1j} z_{i,1k} + 1$**.** The true model includes pairwise interactions within G1, with only 50% of the

possible combinations active ($\gamma_{jk} \in \{1.5, 0, -1.5\}$). A linear GLIMARK assigned similar contributions to G1 and G2 (19.29% vs. 18.08%), failing to distinguish the additional interaction structure in G1 from the main effects shared by both groups. A hybrid GLIMARK using linear and degree-2 polynomial kernels better decomposed these effects, assigning 19.94% to the G1 polynomial component and 6.10% to the G2 linear component. This aligns with the polynomial kernel's feature mapping, which captures two-way interactions without requiring their explicit specification. Logistic regression and its regularized variants also assigned similar contributions to G1 and G2 and therefore did not distinguish the interaction-containing group. Random forest placed somewhat greater importance on G1 than on G2, but the difference was modest and did not provide kernel-specific attribution of the interaction structure. The hybrid GLIMARK model more clearly distinguished G1 as the group containing additional within-group interaction structure, illustrating how the use of complementary kernels can support detection and attribution of potential interactions.

Scenario E: $f(\boldsymbol{z}_i) = \langle \boldsymbol{z}_{i,1}, \boldsymbol{\beta}_1 \rangle + \sum_{j<k} \gamma_{jk} z_{i,1j} z_{i,1k} + 1$**.** We consider only G1, with five active variables. Among the 10 possible two-way interactions, 70% are randomly selected as active. The interaction coefficients satisfy $\gamma_{jk} \in \{1.5, 0, -1.5\}$, with nonzero values corresponding to active interactions, while the main-effect coefficients have magnitude one.

Two hybrid GLIMARK models are evaluated. In both models, a linear kernel is applied to the five individual variables, while either a degree-2 polynomial kernel or an RBF kernel is applied separately to each of the 10 possible pairwise interaction terms. Both models clearly distinguished active from inactive interactions. For the linear-plus-RBF model, active-interaction contributions ranged from approximately 11% to 13%, compared with approximately 4% to 5% for inactive interactions. The linear-plus-polynomial model showed a similar pattern, with active-interaction contributions ranging from approximately 9% to 13%, compared with approximately 3% for inactive interactions.

For the regression and random-forest benchmarks, the five individual predictors and all 10 possible pairwise interaction terms were included as candidate predictors. The regression models also separated active from inactive interactions, while random forest showed weaker separation. Compared with these benchmarks, GLIMARK placed substantially greater relative emphasis on the interaction components and less on the individual-variable components. This pattern is consistent with the simulation design, in which most

candidate interactions were active and their coefficient magnitude exceeded that of the main effects. Thus, Scenario E shows that GLIMARK can identify specific active interaction combinations and attribute a larger share of the fitted signal to the interaction structure.

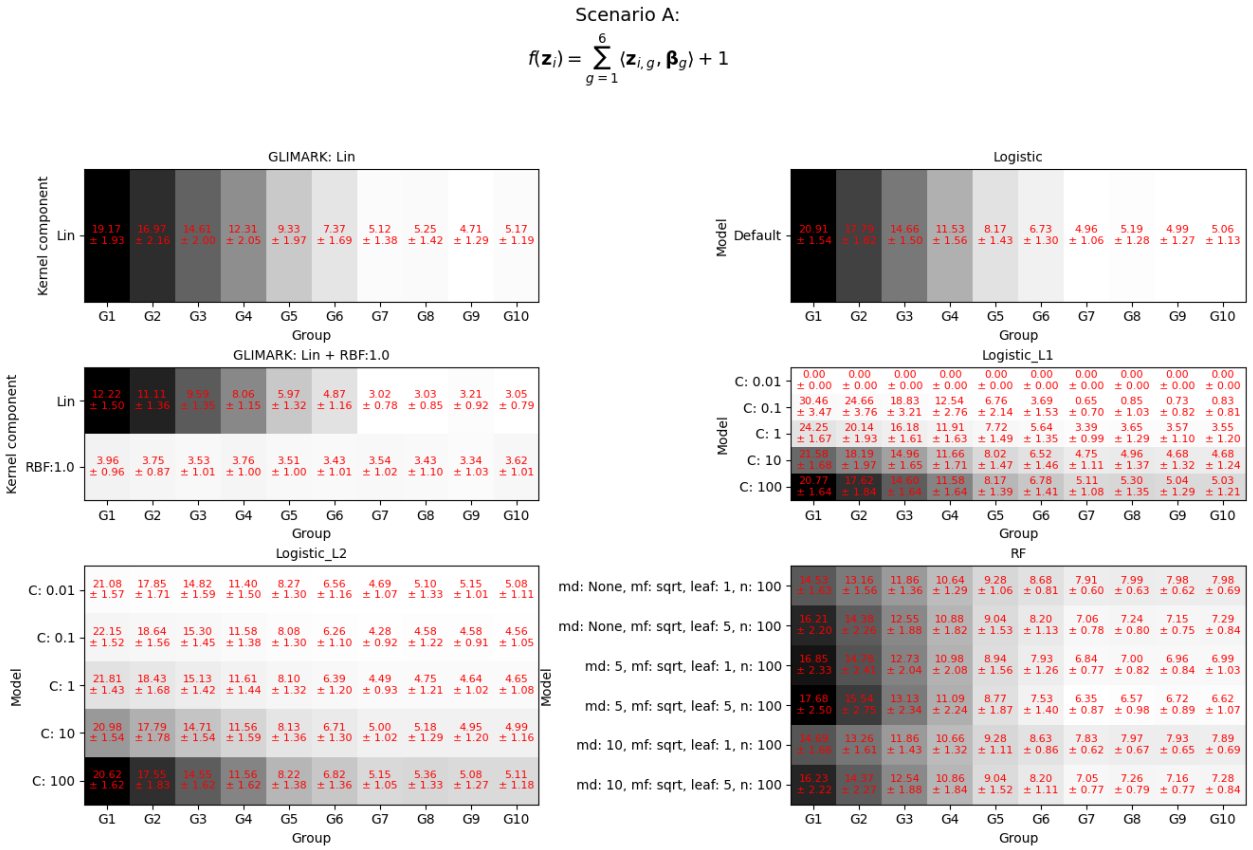


Figure Appendix B.1: Scenario A feature-contribution results.

Scenario B:

$$f(\mathbf{z}_i) = \langle \mathbf{z}^3_{i,1}, \boldsymbol{\beta}_1 \rangle + \langle \mathbf{z}^2_{i,2}, \boldsymbol{\beta}_2 \rangle + \langle \mathbf{z}_{i,1}, \boldsymbol{\beta}_1 \rangle + 1$$

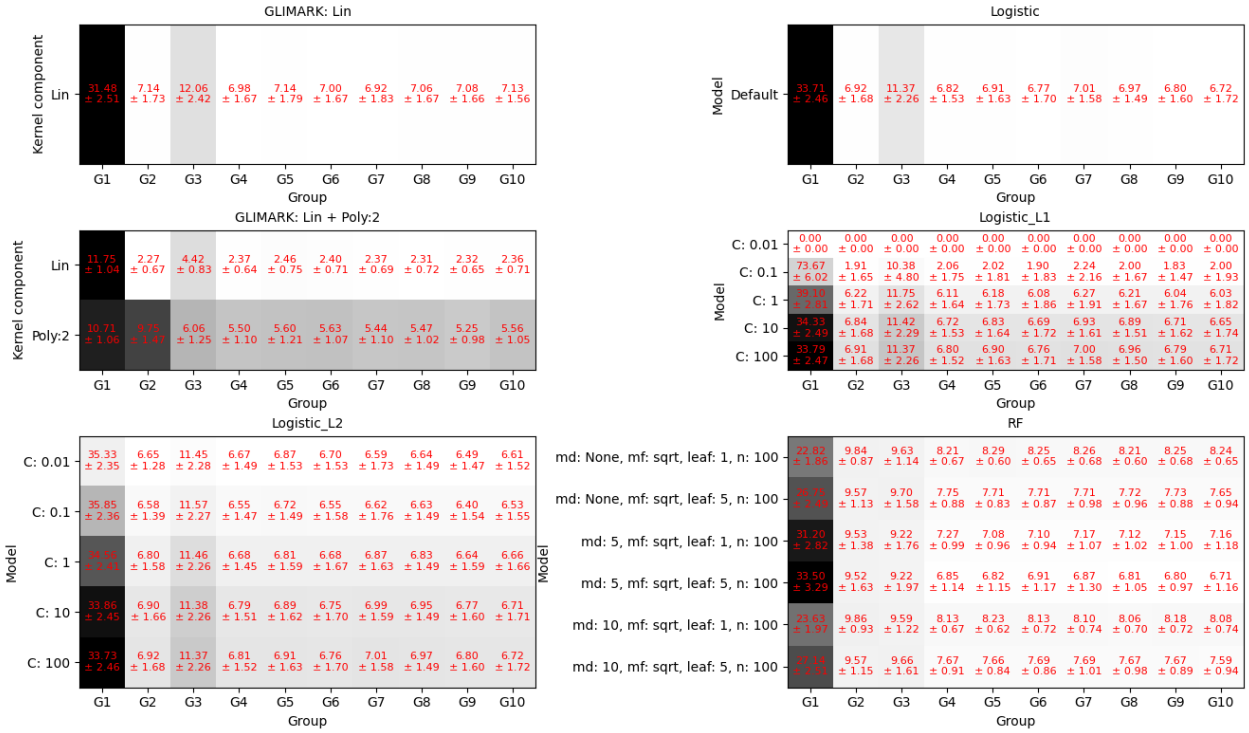


Figure Appendix B.2: Scenario B feature-contribution results.

Scenario C:

$$f(\mathbf{z}_i) = 3 \sum_{g=1}^{6} \langle \mathbf{z}_{i,g}^{\text{exp}}, \boldsymbol{\beta}_g \rangle + \langle \mathbf{z}_{i,4}, \boldsymbol{\beta}_4 \rangle + 1$$

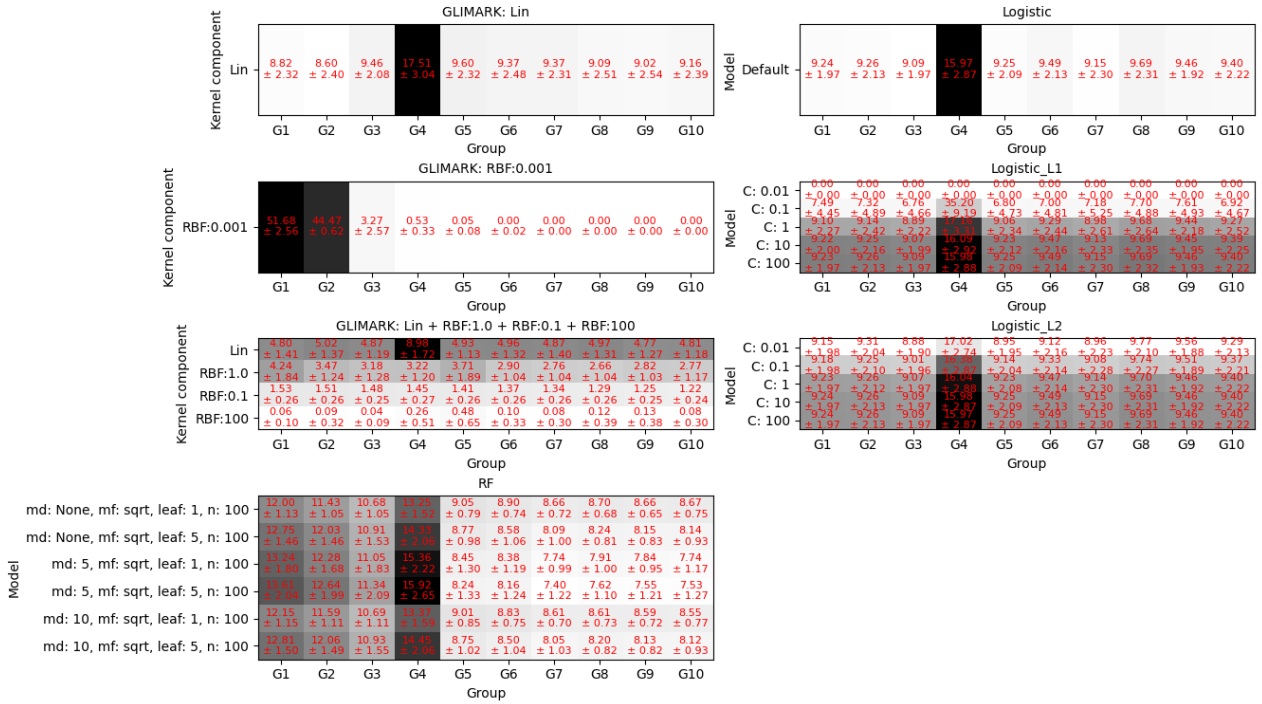


Figure Appendix B.3: Scenario C feature-contribution results.

Scenario D:

$$f(\mathbf{z}_i) = \langle \mathbf{z}_{i,1}, \boldsymbol{\beta}_1 \rangle + \langle \mathbf{z}_{i,2}, \boldsymbol{\beta}_2 \rangle + \sum_{j<k} \gamma_{jk} z_{i,1j} z_{i,1k} + 1$$

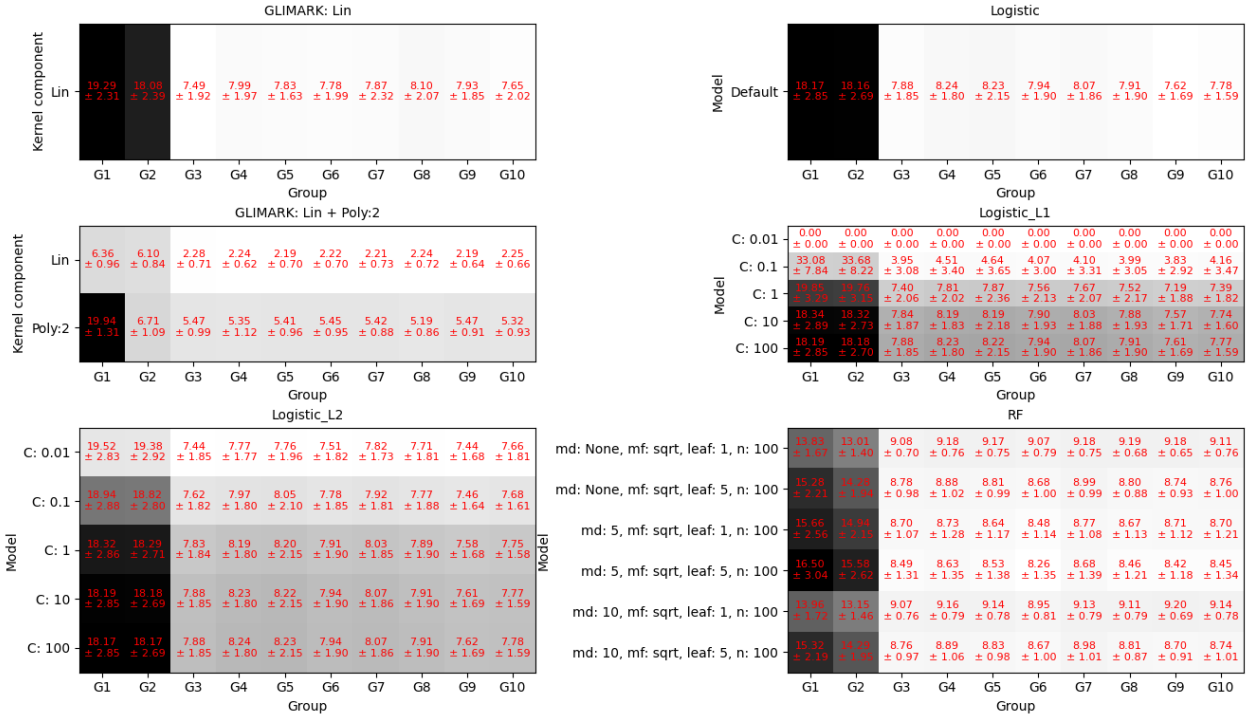


Figure Appendix B.4: Scenario D feature-contribution results.

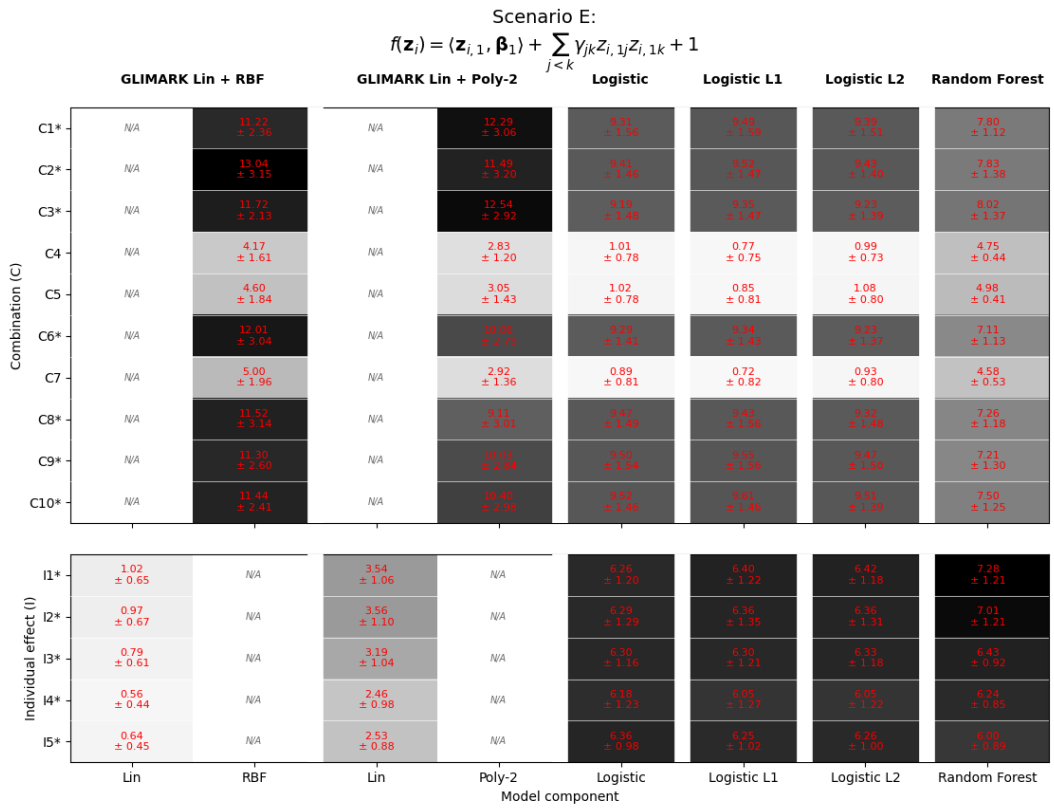


Figure Appendix B.5: Scenario E feature-contribution results.

Summary.

In summary, GLIMARK behaved similarly to regression methods for linear effects, but provided clearer recovery when nonlinear structures were present. Its performance depended on specifying suitable kernels, with hybrid models helping when multiple functional forms occurred together. GLIMARK also uniquely identified groups containing potential within-group interactions and placed greater emphasis on interaction effects than the fully specified benchmark models.

Appendix B.2. Simulation Benchmark Performance

For the second aim, we considered more complex nonlinear data-generating mechanisms designed to better resemble realistic prediction settings, with both group-level and individual-variable effects. Within this setting, we had two related objectives. First, we evaluated the predictive performance of GLIMARK for binary and count outcomes in Scenarios F and G, respectively, and compared it with commonly used benchmark methods. Second, the increased complexity of these scenarios provided a practical setting for evaluating grid-based tuning of λ and T, as would be required in a real-data analysis. Scenario F was evaluated using AUC, accuracy, Youden's J, and F1 score, whereas Scenario G was evaluated using RMSE. The benchmark tuning grids and selected settings are reported in Table Appendix B.1.

Table Appendix B.1: Benchmark tuning grids for simulation Scenarios F and G.

Scenario	Method	Examined settings
F	Logistic regression, ℓ_1 and ℓ_2	$C \in \{0.01, 0.1, 1, 10, 100\}$
F	Linear SVM	$C \in \{0.1, 1, 10\}$
F	RBF SVM	$C \in \{0.1, 1, 10\}$; $\gamma \in \{\text{scale}, 0.1\}$
F	Random forest	Depth $\in$ {unrestricted, 5, 10}; leaf size $\in \{1, 5\}$
F	MLP	Hidden layers $\in \{(64), (128, 64)\}$
F	XGBoost	Depth $\in \{3, 5\}$; learning rate $\in \{0.05, 0.10\}$
F	LightGBM	$(\text{depth}, \text{leaves}) \in \{(3, 15), (5, 31)\}$; learning rate $\in \{0.05, 0.10\}$
G	Poisson regression, ℓ_2	$\alpha \in \{0.001, 0.01, 0.1, 1, 10\}$
G	Ridge regression	$\alpha \in \{0.01, 0.1, 1, 10, 100\}$
G	Linear SVR	$C \in \{0.1, 1, 10\}$; $\epsilon \in \{0.1, 0.5, 1\}$
G	RBF SVR	$C \in \{0.1, 1, 10\}$; $\gamma \in \{\text{scale}, 0.1\}$; $\epsilon \in \{0.1, 0.5, 1\}$
G	Random forest	Depth $\in$ {unrestricted, 5, 10}; leaf size $\in \{1, 5\}$
G	MLP	Hidden layers $\in \{(64), (128, 64)\}$; $\alpha \in \{10^{-4}, 10^{-3}\}$
G	XGBoost–Poisson	Depth $\in \{3, 5\}$; learning rate $\in \{0.05, 0.10\}$
G	LightGBM–Poisson	Depth $\in \{3, 5\}$; leaves $\in \{15, 31\}$; learning rate $= 0.05$

Note: C denotes the inverse regularization parameter, $C = 1/\lambda$. For Poisson and ridge regression, α denotes the weight applied to the regularization penalty, with larger values indicating stronger regularization. For the MLP, α denotes the ℓ_2 penalty applied to the network weights, again with larger values indicating stronger regularization. These symbols follow the notation used by the relevant software packages. The use of α here is an abuse of notation and should not be confused with the GLIMARK representer coefficients α_j defined in the main paper.

Scenario F: $f(\boldsymbol{z}_i) = 3\sin\left(2\sum_{g=1}^{10}|z_{i,g1}| + 1\right) + \exp\left(0.0001\sum_{g=1}^{10} z_{i,g2} - 2\right) + 100\langle \boldsymbol{z}_{i,3}^2, \boldsymbol{\beta}_3\rangle + \langle \boldsymbol{z}_{i,4}, \boldsymbol{\beta}_4\rangle + 1$.

Generally, the AUC of GLIMARK increased with T and decreased with

λ. The best mean AUC and accuracy were both obtained at $T = 180$ and $\lambda = 10^{-7}$, suggesting benefits from relatively weak regularization and a larger number of representers. At this setting, GLIMARK achieved an AUC of 0.801 ± 0.06, an accuracy of 0.801 ± 0.04, a Youden's J of 0.420 ± 0.13, and an F1 score of 0.507 ± 0.12. The best mean Youden's J was 0.421 ± 0.13, attained at $T = 140$ and $\lambda = 10^{-9}$, while the best mean F1 score was 0.515 ± 0.10, attained at $T = 180$ and $\lambda = 10^{-6}$. See Figure Appendix B.6 and Table Appendix B.2.

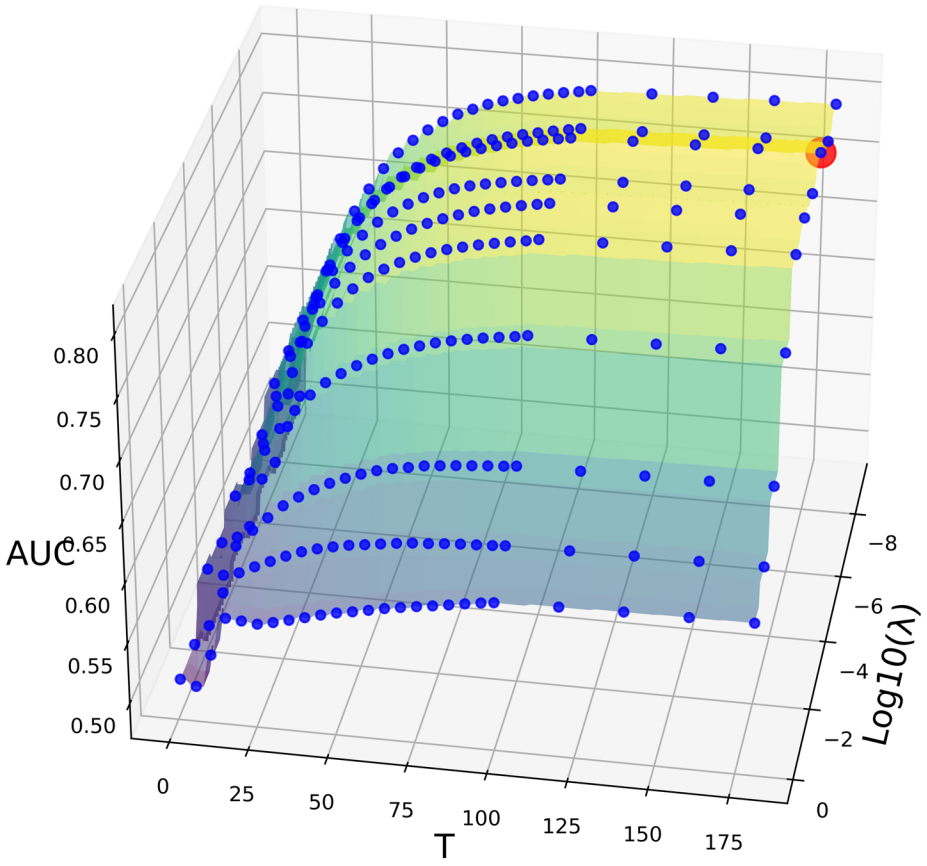


Figure Appendix B.6: Scenario F.

Using AUC as the primary performance criterion, the best GLIMARK configuration achieved an AUC of 0.801 ± 0.06, outperforming the best AUC-selected configurations of logistic regression, linear and RBF SVM, MLP, and random forest. Logistic regression, linear SVM, and MLP had AUC values close to 0.5, while RBF SVM achieved an AUC of 0.539 ± 0.07. Random forest was the strongest nonboosting benchmark, with an AUC of 0.785 ± 0.06. The boosting methods achieved the highest AUC values, with 0.895 ± 0.04 for

XGBoost and 0.910 ± 0.03 for LightGBM. When configurations were instead selected separately by accuracy, Youden's J, and F1, the boosting methods also achieved the highest values for these metrics. GLIMARK remained stronger than the regression, SVM, and MLP benchmarks and was generally comparable to or better than random forest.

Scenario G: $f(\boldsymbol{z}_i) = \frac{1}{20}\big[10\sin\Big(50\sum_{g=1}^{10}|z_{i,g1}|+1\Big)+25\exp\Big(0.1\sum_{g=1}^{10}z_{i,g2}-2\Big)+3\langle \boldsymbol{z}_{i,3}^2, \boldsymbol{\beta}_3\rangle + 5\langle \boldsymbol{z}_{i,4}, \boldsymbol{\beta}_4\rangle + 1\big]$. For the Poisson outcome, GLIMARK showed a different tuning pattern from that observed for the binary outcome in Scenario F. The mean RMSE was minimized at $T = 6$, while adding more representers beyond this point generally reduced predictive performance. Mean RMSE also increased as λ decreased, suggesting that stronger regularization was needed to limit overfitting in this count-data setting. The lowest mean RMSE for GLIMARK was 1.479 ± 0.20, compared with 1.518 ± 0.32 for LightGBM–Poisson, the strongest benchmark, and 1.522 ± 0.33 for XGBoost–Poisson. Thus, GLIMARK provided a modest improvement over the best benchmark model. See Figure Appendix B.7 and Table Appendix B.2.

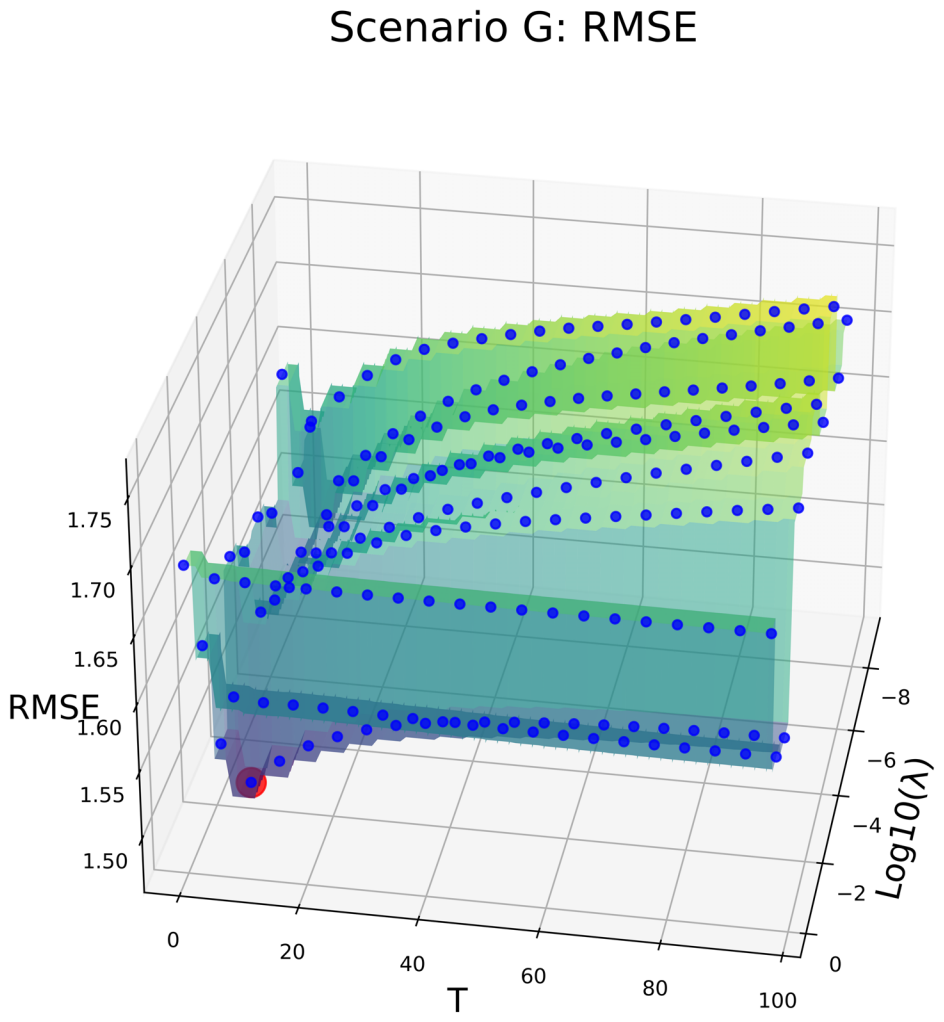


Figure Appendix B.7: Scenario G.

Table Appendix B.2: Predictive performance in simulation Scenarios F and G. For Scenario F, model configurations were selected separately within each model family according to mean AUC, accuracy, Youden's J, and F1. For Scenario G, the reported GLIMARK and benchmark configurations were selected by the lowest mean RMSE within their respective candidate grids. All selections were made over the tuning grids listed in Table Appendix B.1. Results are reported as mean ± standard deviation across 100 simulation replicates.

Model and selected setting	AUC	Accuracy	Youden's J	F1
Scenario F: binary outcome				
Best AUC setting within each model family				
GLIMARK $\lambda = 10^{-7}$, $T = 180$	**0.801 ± 0.06**	0.801 ± 0.04	0.420 ± 0.13	0.507 ± 0.12
Logistic regression, ℓ_1 $C = 0.1$	0.510 ± 0.06	0.547 ± 0.07	0.022 ± 0.11	0.328 ± 0.08
Logistic regression, ℓ_2 $C = 100$	0.510 ± 0.07	0.568 ± 0.06	0.021 ± 0.11	0.313 ± 0.08
Linear SVM $C = 10$	0.510 ± 0.07	0.557 ± 0.05	0.025 ± 0.11	0.324 ± 0.07
RBF SVM $C = 10$, $\gamma =$ scale	0.539 ± 0.07	0.743 ± 0.04	0.000 ± 0.00	0.000 ± 0.00
Random forest depth 5, leaf size 5	0.785 ± 0.06	0.779 ± 0.05	0.340 ± 0.14	0.498 ± 0.12
MLP 64 hidden units	0.510 ± 0.07	0.547 ± 0.07	0.001 ± 0.12	0.307 ± 0.08
XGBoost depth 5, rate 0.05	0.895 ± 0.04	0.784 ± 0.04	0.175 ± 0.11	0.287 ± 0.15
LightGBM depth 3, 15 leaves, rate 0.10	0.910 ± 0.03	0.826 ± 0.04	0.365 ± 0.12	0.519 ± 0.12
Best accuracy setting within each model family				
GLIMARK $\lambda = 10^{-7}$, $T = 180$	0.801 ± 0.06	**0.801 ± 0.04**	0.420 ± 0.13	0.507 ± 0.12
Logistic regression, ℓ_1 $C = 10$	0.510 ± 0.07	0.570 ± 0.06	0.026 ± 0.11	0.315 ± 0.08
Logistic regression, ℓ_2 $C = 10$	0.510 ± 0.07	0.569 ± 0.06	0.024 ± 0.11	0.314 ± 0.08
Linear SVM $C = 1$	0.509 ± 0.07	0.559 ± 0.05	0.027 ± 0.11	0.324 ± 0.07
RBF SVM $C = 10$, $\gamma =$ scale	0.539 ± 0.07	0.743 ± 0.04	0.000 ± 0.00	0.000 ± 0.00
Random forest depth 5, leaf size 5	0.785 ± 0.06	0.779 ± 0.05	0.340 ± 0.14	0.498 ± 0.12
MLP 64 hidden units	0.510 ± 0.07	0.547 ± 0.07	0.001 ± 0.12	0.307 ± 0.08

Continued on next page

Table Appendix B.2 continued from previous page

Model and selected setting	AUC	Accuracy	Youden's J	F1
XGBoost depth 3, rate 0.05	0.894 ± 0.04	0.826 ± 0.04	0.488 ± 0.12	0.618 ± 0.09
LightGBM depth 3, 15 leaves, rate 0.05	0.903 ± 0.03	0.841 ± 0.04	0.563 ± 0.10	0.671 ± 0.08
Best Youden's J setting within each model family				
GLIMARK $\lambda = 10^{-9}$, $T = 140$	0.797 ± 0.06	0.799 ± 0.04	$\mathbf{0.421 \pm 0.13}$	0.501 ± 0.12
Logistic regression, ℓ_1 $C = 10$	0.510 ± 0.07	0.570 ± 0.06	0.026 ± 0.11	0.315 ± 0.08
Logistic regression, ℓ_2 $C = 10$	0.510 ± 0.07	0.569 ± 0.06	0.024 ± 0.11	0.314 ± 0.08
Linear SVM $C = 0.1$	0.508 ± 0.07	0.556 ± 0.06	0.032 ± 0.12	0.330 ± 0.08
RBF SVM $C = 0.1$, $\gamma = \text{scale}$	0.535 ± 0.07	0.700 ± 0.06	0.020 ± 0.07	0.140 ± 0.12
Random forest depth 5, leaf size 1	0.782 ± 0.06	0.775 ± 0.05	0.361 ± 0.14	0.514 ± 0.11
MLP 128 and 64 hidden units	0.502 ± 0.07	0.525 ± 0.09	0.001 ± 0.10	0.313 ± 0.08
XGBoost depth 3, rate 0.05	0.894 ± 0.04	0.826 ± 0.04	0.488 ± 0.12	0.618 ± 0.09
LightGBM depth 3, 15 leaves, rate 0.05	0.903 ± 0.03	0.841 ± 0.04	0.563 ± 0.10	0.671 ± 0.08
Best F1 setting within each model family				
GLIMARK $\lambda = 10^{-6}$, $T = 180$	0.801 ± 0.06	0.797 ± 0.04	0.415 ± 0.12	$\mathbf{0.515 \pm 0.10}$
Logistic regression, ℓ_1 $C = 0.1$	0.510 ± 0.06	0.547 ± 0.07	0.022 ± 0.11	0.328 ± 0.08
Logistic regression, ℓ_2 $C = 1$	0.509 ± 0.07	0.568 ± 0.06	0.022 ± 0.10	0.314 ± 0.07
Linear SVM $C = 0.1$	0.508 ± 0.07	0.556 ± 0.06	0.032 ± 0.12	0.330 ± 0.08
RBF SVM $C = 1$, $\gamma = \text{scale}$	0.535 ± 0.07	0.700 ± 0.06	0.020 ± 0.07	0.140 ± 0.12

Continued on next page

Table Appendix B.2 continued from previous page

Model and selected setting	AUC	Accuracy	Youden's J	F1
Random forest depth 5, leaf size 1	0.782 ± 0.06	0.775 ± 0.05	0.361 ± 0.14	0.514 ± 0.11
MLP 128 and 64 hidden units	0.502 ± 0.07	0.525 ± 0.09	0.001 ± 0.10	0.313 ± 0.08
XGBoost depth 3, rate 0.05	0.894 ± 0.04	0.826 ± 0.04	0.488 ± 0.12	0.618 ± 0.09
LightGBM depth 3, 15 leaves, rate 0.05	0.903 ± 0.03	0.841 ± 0.04	0.563 ± 0.10	0.671 ± 0.08

Scenario G: count outcome

Model	**Selected setting**	**Test RMSE**
GLIMARK	$\lambda = 0.01,\ T = 6$	$\mathbf{1.479 \pm 0.20}$
Poisson regression, ℓ_2	$\alpha = 1$	1.600 ± 0.30
Ridge regression	$\alpha = 100$	1.647 ± 0.28
Linear SVR	$C = 0.1,\ \epsilon = 0.5$	1.666 ± 0.29
RBF SVR	$C = 1,\ \epsilon = 1,\ \gamma = \text{scale}$	1.601 ± 0.31
Random forest	100 trees; unrestricted depth; leaf size 5; square-root feature sampling	1.585 ± 0.31
MLP	$(128, 64)$; $\alpha = 10^{-4}$; learning rate $= 10^{-3}$	1.706 ± 0.30
XGBoost–Poisson	100 trees; depth 3; learning rate $= 0.05$	1.522 ± 0.33
LightGBM–Poisson	100 trees; depth 3; 15 leaves; learning rate $= 0.05$	1.518 ± 0.32

Note: Numerical results are rounded to three decimal places; therefore, different selected settings may occasionally display identical performance values after rounding.

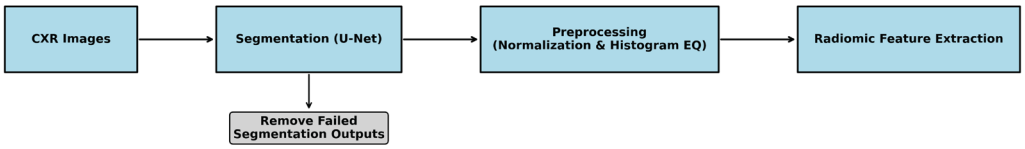
Summary. GLIMARK achieved strong predictive performance in both simulation settings. For the binary outcome in Scenario F, its AUC was higher than those of logistic regression, SVM, MLP, and random forest, although XGBoost and LightGBM achieved the highest AUCs. GLIMARK also showed competitive accuracy, Youden's J, and F1 performance across its metric-specific selected configurations. For the count outcome in Scenario G, GLIMARK achieved the lowest RMSE among all evaluated methods, outperforming both boosting models and the remaining benchmarks.

Appendix C. Real Data Analysis

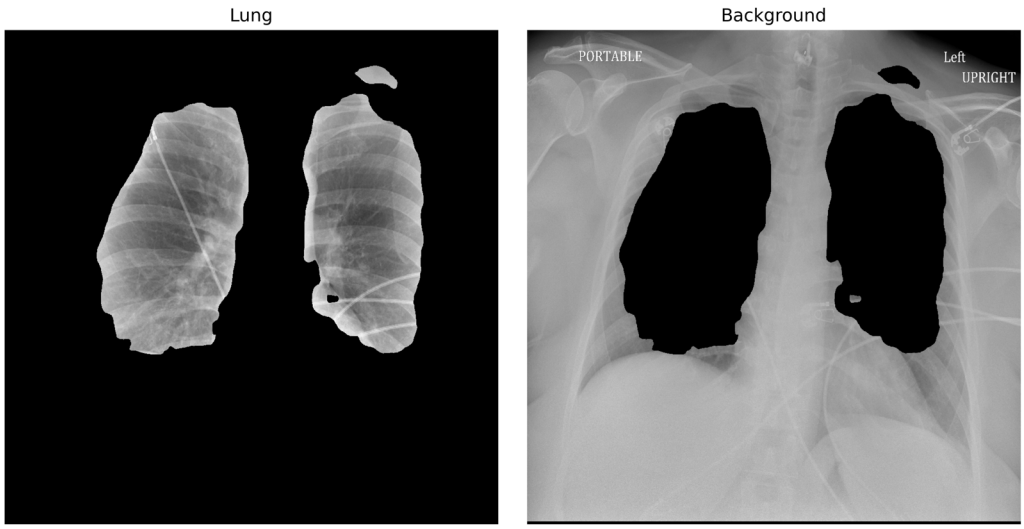
Appendix C.1. Patient Population and Radiomic Features

We analyzed 2,396 COVID-19 patients from the Michigan Medicine Precision Health database with at least one AP or PA view CXR between March 2020 and March 2022. The binary outcome was ICU admission within 30 days of diagnosis. Median age was 60 (IQR: 44–77), 54% were male, and 31.5% required ICU care. This study was approved by the Institutional Review Board (IRB) of University of Michigan.

Only one CXR per patient, taken on the date of first COVID-19 diagnosis, was included. Images (DICOM format, 2757×3195 pixels) were normalized to [0, 255] and contrast-enhanced via histogram equalization. Lung segmentation was performed using a U-net trained on public datasets with manual masks: the Montgomery County dataset and a 55-image subset from Indiana University (Ronneberger et al., 2015; Candemir et al., 2013; Xue et al., 2023). Radiomic features were extracted using PyRadiomics (Van Griethuysen et al., 2017). Each image was enhanced using one of eleven filters, followed by feature extraction across seven predefined classes, for both lung and background segmentations. This yielded 2,046 features per image. See Figure Appendix C.1 for details.



(a) Flowchart of image processing.



(b) An example of image segmentation.

Figure Appendix C.1: Data processing pipeline.

Appendix C.2. Hierarchical Model Development and Bootstrap Stability Selection

The full dataset was randomly divided into an 80% development set and a 20% holdout set. The hierarchical model was developed sequentially from coarse to fine feature resolution using only the development set, following the workflow shown in Figure Appendix C.2. At each stage, all components under consideration formed the candidate model for that level. The most stable kernels or feature views were then selected to finalize M1, M2, and M3, with each finalized model carried forward as the base for introducing components from the next level. For the first two stages, selection stability and uncertainty were evaluated using 100 bootstrap resamples of the development set, with components ranked according to their contribution within each resample. The 20% holdout set was not used during model development

or hierarchical component selection and was reserved exclusively for the final evaluation of the completed models.

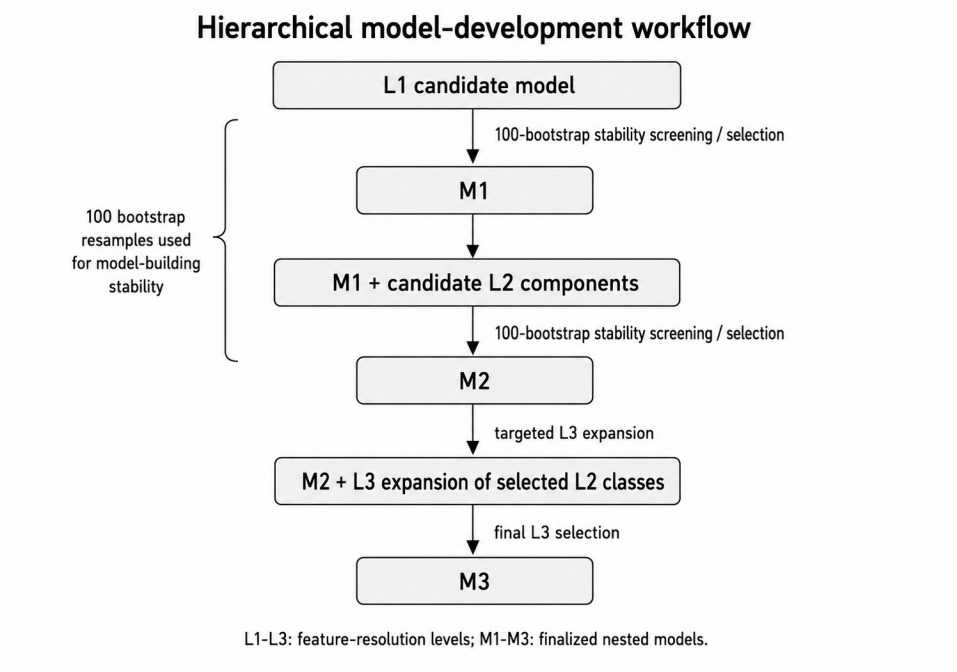
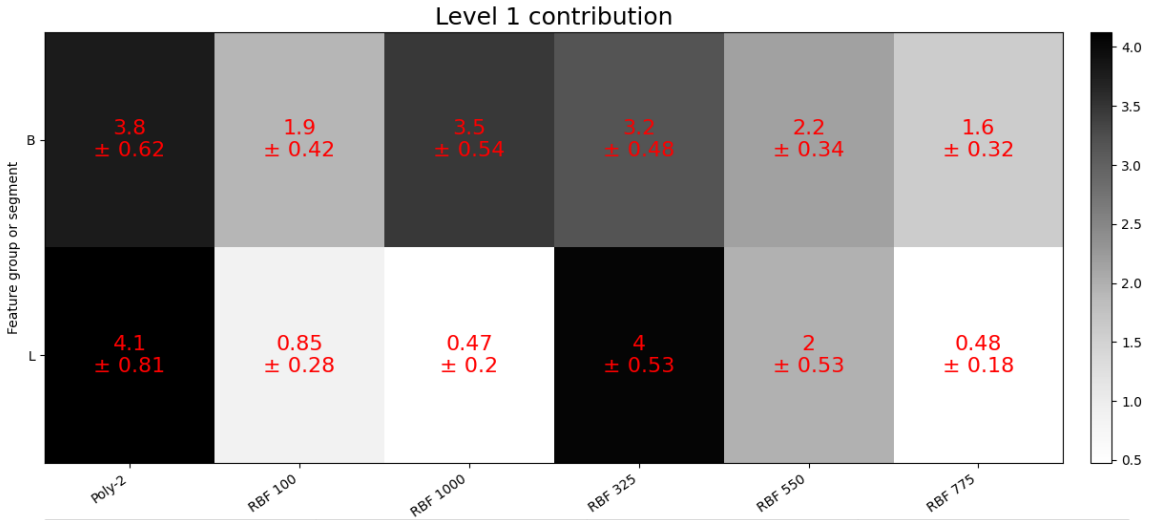


Figure Appendix C.2: Hierarchical model-development workflow. Candidate components were evaluated sequentially across three feature-resolution levels. Bootstrap stability screening was used to finalize M1 and M2, followed by targeted Level-3 expansion and final selection of M3.

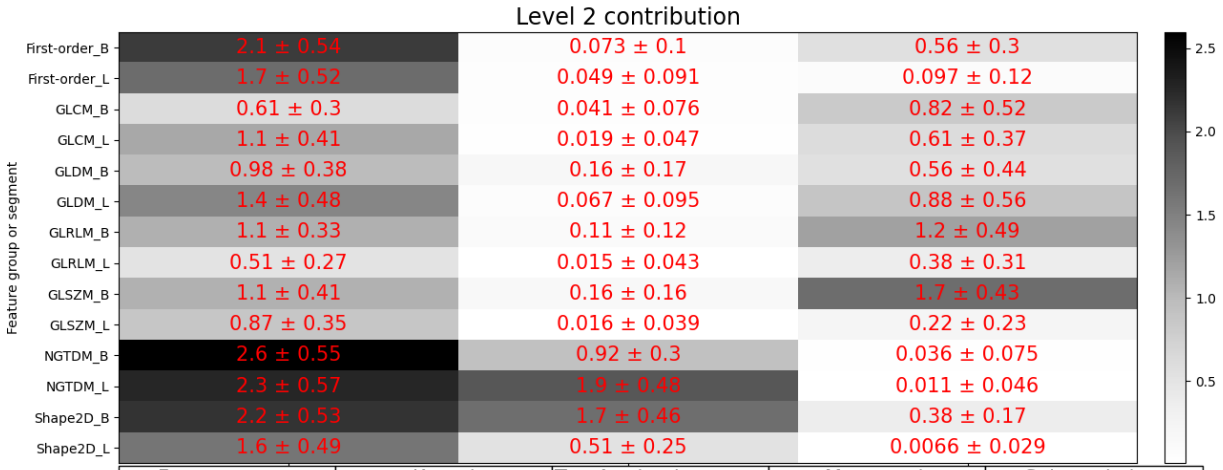
At Level 1, only the lung and background compartments were available, so the main objective was to identify a compact set of kernels rather than separate compartment–kernel pairs. We considered a degree-2 polynomial kernel and RBF kernels spanning a broad range of bandwidths appropriate for the segmentation-level representation. As shown in Figure Appendix C.3, the polynomial kernel and the RBF kernel with $\sigma = 325$ were ranked among the top two kernels in all 100 bootstrap resamples, with mean ranks of 1.31 and 1.69, respectively. These two kernels were therefore retained for both compartments to define M1.



Kernel	Top-2 selection count	Mean rank	Selected kernel
Poly-2	100	1.31	True
RBF 325	100	1.69	True
RBF 550	0	3.46	False
RBF 1000	0	3.65	False
RBF 100	0	5.06	False
RBF 775	0	5.83	False

Figure Appendix C.3: Level-1 contribution and kernel-selection results across 100 bootstrap resamples. Red cell values show the mean sum of the absolute representer coefficients, together with its standard deviation, rather than contribution percentages. The table summarizes the frequency with which each kernel appeared among the top two kernels and its mean rank. At Level 1, ranking was based on each kernel's marginal contribution summed over the background and lung compartments, because kernel selection, rather than selection of a specific compartment–kernel view, was the primary objective at this stage.

Starting from M1, we added candidate Level-2 feature-class views and used the degree-2 polynomial kernel together with smaller RBF bandwidths appropriate for the more granular feature-class representations. Because Level 2 contained many feature-class–compartment combinations, selection was performed at the individual view level. As shown in Figure Appendix C.4, the four most stable views were NGTDM_B–Poly-2, NGTDM_L–Poly-2, Shape2D_B–Poly-2, and First-order_B–Poly-2, which appeared among the top four views in 86, 64, 63, and 53 of the 100 bootstrap resamples, respectively. These views were retained to define M2. Their corresponding Level-2 classes were then expanded into individual features, and this targeted Level-3 expansion was combined with M2 to produce the final model M3.



Feature group	Kernel	Top-4 selection count	Mean rank	Selected view
NGTDM_B	Poly-2	86	2.77	True
NGTDM_L	Poly-2	64	4.26	True
Shape2D_B	Poly-2	63	4.49	True
First-order_B	Poly-2	53	5.03	True
NGTDM_L	RBF 10	37	6.35	False
Shape2D_B	RBF 10	24	8.08	False
First-order_L	Poly-2	20	8.1	False
Shape2D_L	Poly-2	17	8.7	False
GLSZM_B	RBF 150	13	7.92	False
GLDM_L	Poly-2	10	10.35	False

Figure Appendix C.4: Level-2 contribution and view-selection results across 100 bootstrap resamples. Red cell values show the mean sum of the absolute representer coefficients, together with its standard deviation, rather than contribution percentages. The table summarizes the frequency with which each view appeared among the top four views and its mean rank.

Appendix C.3. Performance

Table Appendix C.1: Predictive performance on the 80% development set and the 20% holdout set. Based on the simulation results, the best-performing GLIMARK setting and the best-performing setting within each benchmark family were selected and then fixed for the real-data analysis. Cross-validation results are reported as mean ± standard deviation across five folds.

Model	AUC	Accuracy	Youden's J	F1
80% development set: 5-fold cross-validation				
M1	0.799 ± 0.04	0.743 ± 0.04	0.464 ± 0.07	0.631 ± 0.06
M2	0.824 ± 0.04	$\mathbf{0.762 \pm 0.03}$	$\mathbf{0.500 \pm 0.07}$	$\mathbf{0.655 \pm 0.06}$
M3	$\mathbf{0.832 \pm 0.02}$	0.756 ± 0.03	0.487 ± 0.06	0.646 ± 0.05
Logistic ℓ_1	0.810 ± 0.02	0.759 ± 0.02	0.480 ± 0.06	0.642 ± 0.05
Logistic ℓ_2	0.753 ± 0.03	0.735 ± 0.02	0.399 ± 0.05	0.588 ± 0.04
Random Forest	0.824 ± 0.04	0.764 ± 0.03	0.493 ± 0.06	0.650 ± 0.05
RBF-SVM	0.821 ± 0.03	0.710 ± 0.02	0.467 ± 0.05	0.634 ± 0.04
XGBoost	0.839 ± 0.03	0.782 ± 0.01	0.367 ± 0.03	0.545 ± 0.03
LightGBM	0.837 ± 0.03	0.777 ± 0.02	0.517 ± 0.05	0.666 ± 0.04
MLP	0.802 ± 0.01	0.734 ± 0.01	0.443 ± 0.03	0.617 ± 0.02

20% holdout set

Model	AUC	Accuracy	Youden's J	F1
M1	0.784	0.736	0.452	0.625
M2	0.825	**0.780**	0.520	**0.669**
M3	**0.835**	0.751	**0.524**	0.667
Logistic ℓ_1	0.801	0.755	0.462	0.631
Logistic ℓ_2	0.755	0.736	0.398	0.588
Random Forest	0.820	0.763	0.507	0.659
RBF-SVM	0.796	0.692	0.406	0.600
XGBoost	0.830	0.809	0.448	0.619
LightGBM	0.831	0.782	0.484	0.646

Continued on next page

Table Appendix C.1 continued from previous page

Model	AUC	Accuracy	Youden's J	F1
MLP	0.773	0.765	0.459	0.629

Across the 80% development set, the mean five-fold cross-validation results showed a clear improvement from M1 to M2 across all reported metrics. M3 further increased the mean AUC from 0.824 for M2 to 0.832, approaching the best-performing benchmark models, XGBoost and LightGBM, with mean AUCs of 0.839 and 0.837, respectively. In contrast, M2 retained slightly higher mean accuracy, Youden's J, and F1 score than M3, indicating that the Level-3 expansion primarily improved discrimination rather than classification at the selected cutoff. On the independent 20% holdout set, M3 achieved the highest AUC among all evaluated models (0.835) and the highest Youden's J (0.524), while M2 produced slightly higher accuracy and F1 score than M3. Overall, M2 provided the strongest balanced classification performance among the GLIMARK models, whereas M3 provided the strongest discrimination and the best performance on the independent holdout set in terms of AUC and Youden's J.

Appendix C.4. Nyström Comparison

One remaining question is how much information is lost when the Nyström approximation is used in place of the full kernel representation. To assess this, we conducted a sensitivity analysis using $d = 100$ uniformly sampled Nyström landmarks. Agreement between the representer rankings obtained from the original GLIMARK procedure and the Nyström approximation was evaluated for M1–M3 fitted to the full dataset, as these models were used for the final feature-importance analysis. Rank-biased overlap (RBO) was calculated at $p = 0.90$, 0.95, and 0.98. Smaller values of p place greater emphasis on agreement among the highest-ranked representers, whereas larger values distribute the weight more evenly across the full ranking. These results are reported in Table Appendix C.2. The corresponding loss in predictive performance was evaluated on the 20% holdout set and is reported in Table Appendix C.3.

Overall, the Nyström approximation had only a limited effect on the representer rankings obtained from the original GLIMARK procedure. Agreement was particularly strong when greater emphasis was placed on the highest-ranked representers. Although some disagreement remained among lower-

Table Appendix C.2: Rank-biased overlap between the representer rankings obtained from the original GLIMARK procedure and the Nyström approximation with $d = 100$ uniformly sampled landmarks.

Model	$p = 0.90$	$p = 0.95$	$p = 0.98$
M1	0.861	0.796	0.727
M2	0.842	0.773	0.717
M3	0.840	0.770	0.713

Table Appendix C.3: Predictive performance of the original GLIMARK procedure and the Nyström approximation with $d = 100$ uniformly sampled landmarks on the 20% holdout test set.

Metric	Model	Original	Nyström	Δ
AUC	M1	0.7840	0.7762	−0.0078
	M2	0.8250	0.8191	−0.0059
	M3	0.8350	0.8284	−0.0066
Accuracy	M1	0.7360	0.7263	−0.0097
	M2	0.7800	0.7741	−0.0059
	M3	0.7510	0.7484	−0.0026
Youden's J	M1	0.4520	0.4495	−0.0025
	M2	0.5200	0.5142	−0.0058
	M3	0.5240	0.5187	−0.0053
F1 score	M1	0.6250	0.6231	−0.0019
	M2	0.6690	0.6638	−0.0052
	M3	0.6670	0.6614	−0.0056

ranked and less influential representers, the approximation did not materially alter the ranking of the most important representers identified by GLIMARK. Uniform sampling was used as a standard default approach in this analysis, although alternative landmark-selection strategies have been proposed, including data-dependent nonuniform sampling schemes (Drineas et al., 2005). Meanwhile, the number of landmarks d may be chosen according to the sample size N and the desired balance between computational efficiency and approximation accuracy, we did not attempt an exhaustive search over all sampling methods and values of d. Rather, this analysis was intended to provide a representative sensitivity assessment under one practical Nyström setting.

Appendix C.5. Model Comparison and Feature Importance

For feature importance, we consider Model 3 (M3) the best model due to its overall performance. We refit M3 and the benchmark models with readily interpretable feature-importance measures using the entire dataset; therefore, SVM, MLP, XGBoost, and LightGBM were not included in this comparison. Figure Appendix C.5 shows the feature importance across different models. Level 2 was used as the common resolution for this comparison. Level 1 was too coarse for feature-class interpretation, whereas a Level 3 comparison would not be uniform across models: the benchmark models used all individual features, while GLIMARK expanded only the four feature classes selected during hierarchical development into individual features. Aggregation at Level 2 therefore provided the most comparable feature-class summary across models.

The behavior of Model 3 is close to that of lasso-regularized logistic regression in that it highlights fewer, more important feature classes. GLIMARK further focuses on three dominant classes: First-Order, NGTDM, and Shape2D. These findings are consistent with existing literature. First-Order radiomic features have been shown to predict ICU admission and intubation in COVID-19 patients (Varghese et al., 2021), aligning with findings that increased pixel intensity in lung regions corresponds to COVID-19-related lung involvement (Al-Zyoud et al., 2023). Shape2D_L features measure the shape of lung area, and it is shown that severe COVID-19 patients have deformities to the surfaces of the lungs (Hiremath et al., 2024). Previous studies have not extensively reported NGTDM features in COVID-19 prognosis, but our analysis found them to be significant predictors as well.

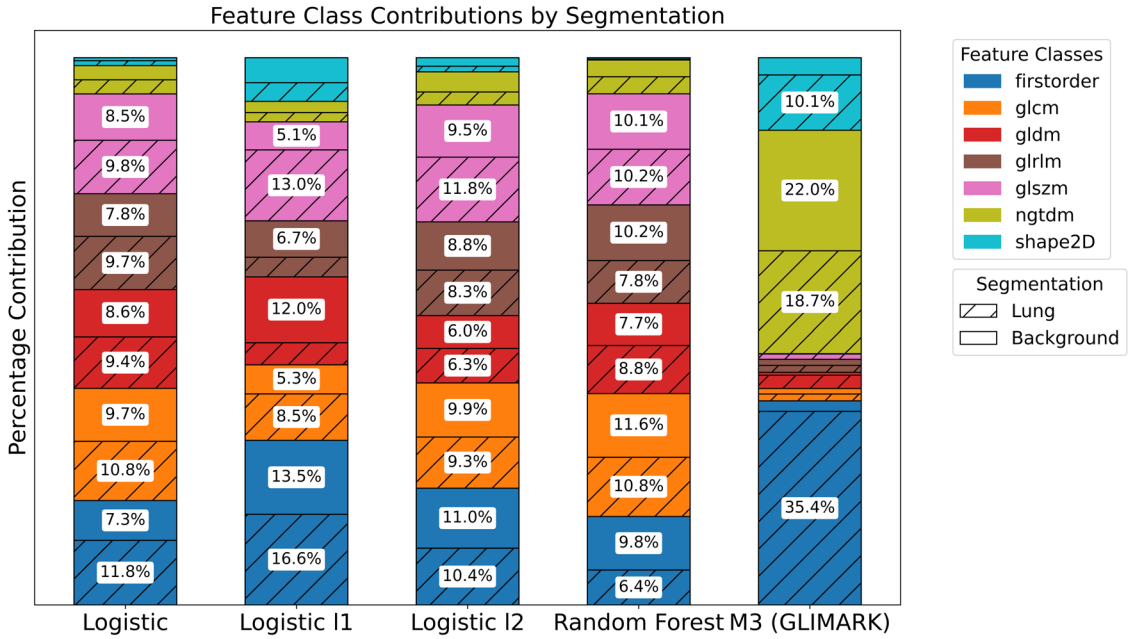
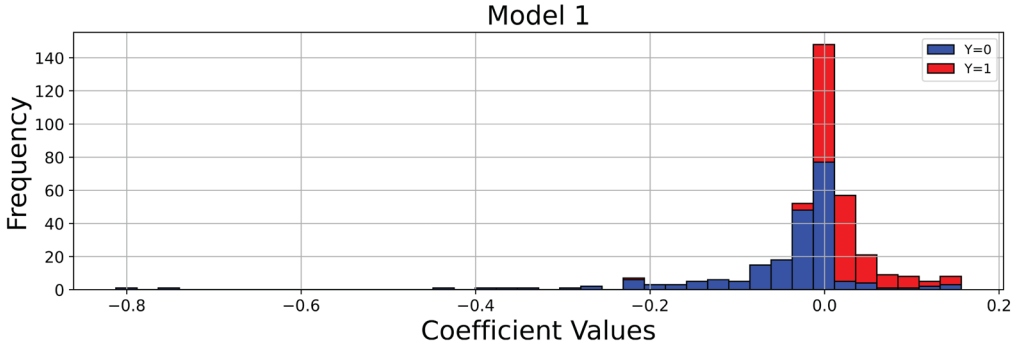


Figure Appendix C.5: Feature importance summarized at the Level 2 feature-class resolution for comparability across models. Contributions for all models were aggregated using the corresponding contribution metrics defined in the simulation section.

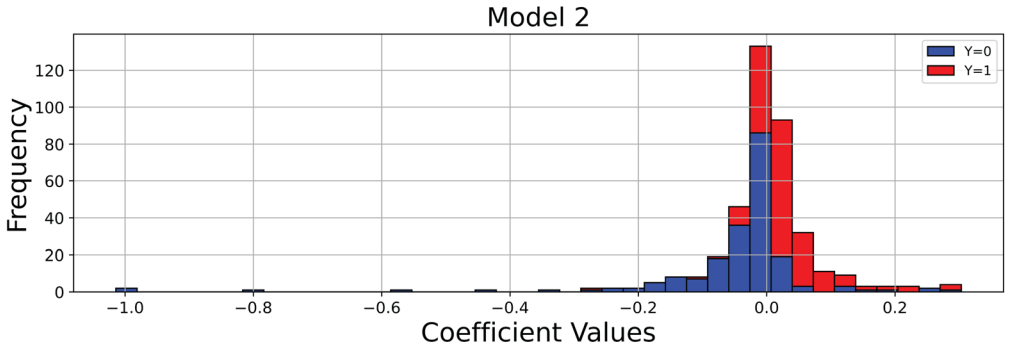
Appendix C.6. Representers

Even we consider M3 the best model for feature importance, in this section, we revisit all three models to highlight their distinct and complementary findings. The GLIMARK model's interpretability stems from its similarity to existing data points, i.e., positive α coefficients are expected to correlate with ICU escalation cases. To assess this, we compared the signs of the $\hat{\boldsymbol{\alpha}}$ coefficients with the true labels of the training points for the three models in Figure Appendix C.6. With 400 representers, Model 1 had the greatest agreement rate (93%), followed by Model 2 (87%). Model 3 only had 60% agreement rate. One explanation is that both cosine similarity and Euclidean similarity provide limited interpretability when comparing two points based on a single attribute.

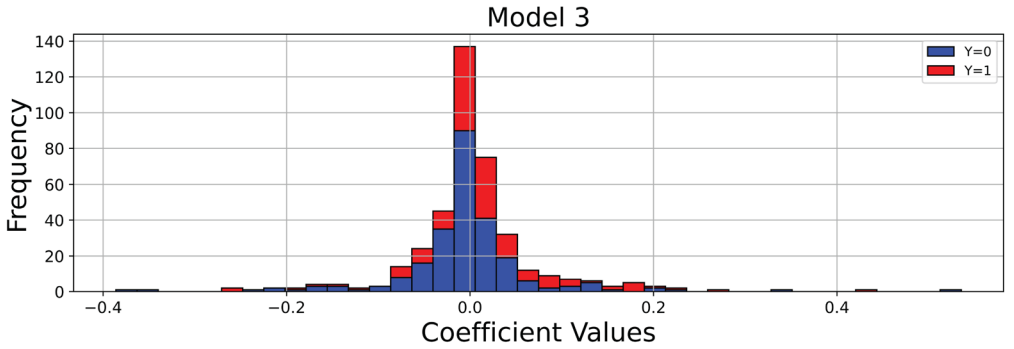
Another interesting finding was that the top representers varied depending on the scale of the data. Figure Appendix C.6 shows that in Model 1, patient 1417 is the most influential across both segments. However, when features are further categorized by classes in Model 2, patient 1417 is no longer among



Coefficient	Patient	Feature	Kernel	Y
-0.81	individual1417	L	RBF: 325.0	0.0
-0.74	individual1417	L	RBF: 550.0	0.0
0.66	individual1567	B	RBF: 1000.0	1.0



Coefficient	Patient	Feature	Kernel	Y
-1.01	individual1718	ngtdm_L	RBF: 10.0	0.0
-1.0	individual72	shape2D_B	Poly: 2	0.0
-0.79	individual1267	glrlm_B	RBF: 150.0	0.0



Coefficient	Patient	Feature	Kernel	Y
0.54	individual2218	original_shape2...	Poly: 2	0.0
0.44	individual783	wavelet-HL_firs...	Poly: 2	1.0
-0.39	individual1698	wavelet-HL_firs...	Poly: 2	0.0

Figure Appendix C.6: Coefficient contribution and top representers. For each model only the three most influential points are displayed.

the top ten representers; instead, patient 1718 with NGTDM_L emerges as the most influential. Similarly, transitioning from Model 2 to Model 3 alters the most influential patients, emphasizing the impact of individual feature selection.

Since kernel selection and feature aggregation methods are not unique, our analysis of CXR images is exploratory rather than definitive, focusing on potential relationships between image features and ICU outcomes. To support this, we present CXR images for the top representers in each model in Figure Appendix C.7 , aiming to guide future research by professional radiologists and other interested researchers in interpreting visual features linked to ICU escalation.

Appendix C.7. Computation

Without the Nyström approximation, GLIMARK requires evaluating kernel matrices of size $N \times N$ for each of the $P \times Q$ views, resulting in a total kernel construction cost of order $\mathcal{O}(N^2PQ)$. To reduce this burden, we approximate each kernel block using d landmark columns, so that each $N \times N$ block is replaced by an $N \times d$ representation. This reduces storage and kernel construction from order $\mathcal{O}(N^2PQ)$ to $\mathcal{O}(dNPQ)$, where $d \ll N$. Separately, the column generation algorithm selects only a small number T of terms in the final model, so the optimization is carried out on a sparse active set rather than all NPQ candidate columns. Moreover, kernel computations are parallelizable across views and highly amenable to GPU acceleration. Because of the moderate real-data sample size, we tested the original M3 model using the full kernels without the Nyström approximation. For $N =$ 2,396 and $PQ \approx 300$, end-to-end training required approximately 25 minutes on an Apple M1 MacBook Pro using MPS and approximately 20 minutes on a desktop computer equipped with an NVIDIA RTX 3080 Ti GPU.

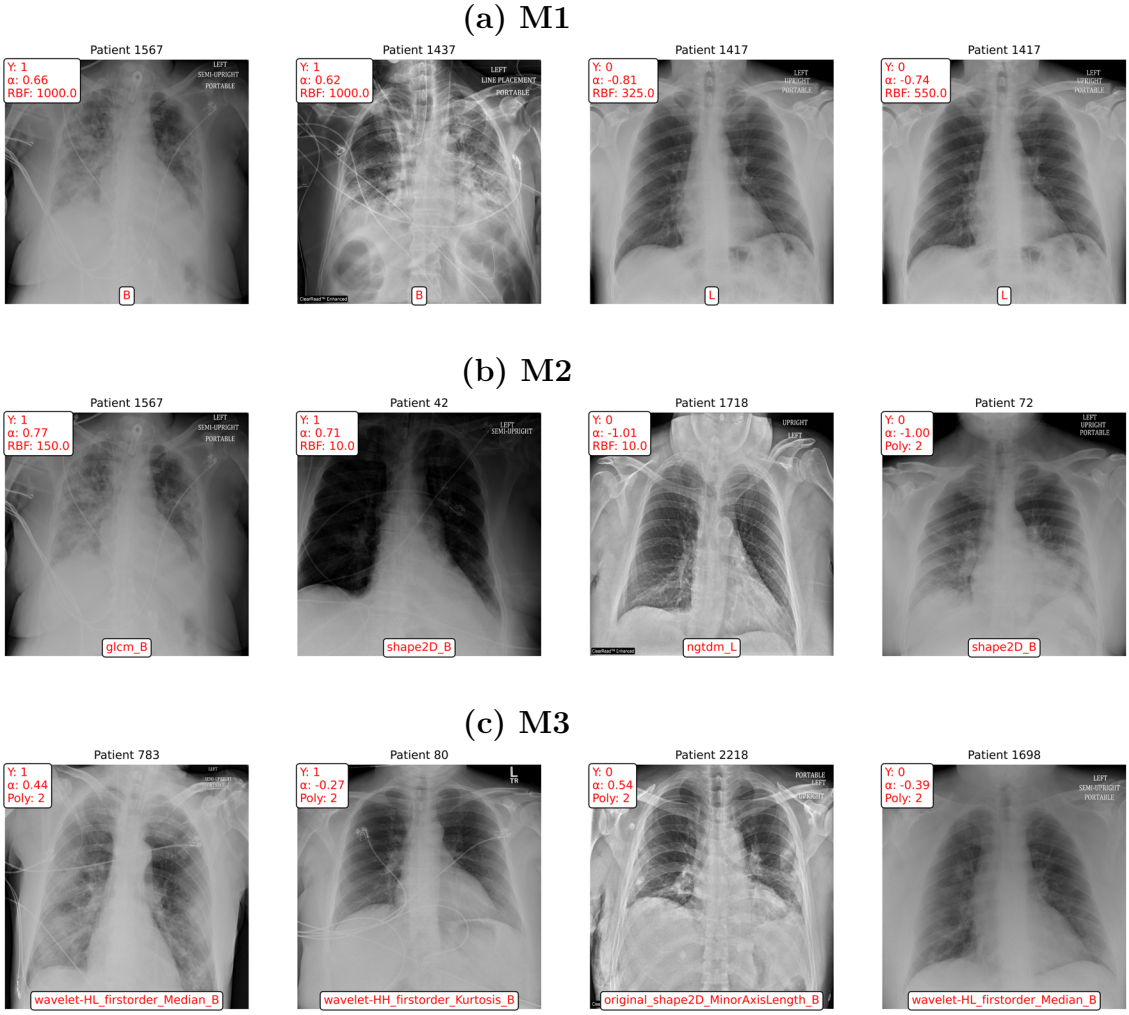


Figure Appendix C.7: Top two cases and controls from each model represent the most influential patients—those most predictive of ICU escalation; cases resemble high-risk patients and controls resemble low-risk ones.

Appendix D. Radiomic Features

Table Appendix D.1: Overview of filter types and feature classes used in the radiomics feature extraction.

Filter Type	**Filter Explanation**	**Feature Class**	**Feature Class Explanation**
Exponential	Applies the exponential of absolute intensity values, then rescales to the original image range.	First Order	Describes basic statistics of intensity values, such as mean, variance, skewness, and entropy.
Gradient	Calculates the gradient magnitude to emphasize edges and regions with rapid intensity change.	GLCM (Gray Level Co-occurrence Matrix)	Describes texture by measuring how often pairs of intensity levels occur at specific distances and angles.
LBP-2D (Local Binary Patterns)	Encodes local texture by thresholding each pixel against its neighbors in 2D.	GLDM (Gray Level Dependence Matrix)	Captures the degree to which neighboring voxels are similar in intensity, reflecting localized texture patterns.
Logarithm	Applies the logarithm of absolute intensity plus one, then rescales to the original image range.	GLRLM (Gray Level Run Length Matrix)	Quantifies contiguous runs of the same intensity along specified directions to assess texture smoothness or granularity.
Original	Keeps the raw image unchanged, with no filter applied.	GLSZM (Gray Level Size Zone Matrix)	Measures connected regions (zones) of the same intensity and quantifies their size to characterize coarse or fine textures.
Square	Squares the intensity values and rescales them, preserving the sign of the original values.	NGTDM (Neighborhood Gray Tone Difference Matrix)	Describes contrast and coarseness by comparing each voxel's intensity to the average of its neighborhood.
Square Root	Computes the square root of absolute intensity, then restores original signs and rescales.	Shape2D	Captures geometric features of 2D regions, including area, perimeter, and axis lengths.
Wavelet-HH, HL, LH, LL	Decomposes the image using discrete wavelet transform into high- and low-frequency subbands.		

Note: For detailed definitions of each radiomic feature, refer to Van Griethuysen et al. (2017).

References

Al-Zyoud, W., Erekat, D., Saraiji, R., 2023. COVID-19 chest X-ray image analysis by threshold-based segmentation. Heliyon 9.

Bi, J., Zhang, T., Bennett, K.P., 2004. Column-generation boosting methods for mixture of kernels, in: Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 521–526.

Candemir, S., et al., 2013. Lung segmentation in chest radiographs using anatomical atlases with nonrigid registration. IEEE Transactions on Medical Imaging 33, 577–590.

Drineas, P., Mahoney, M.W., Cristianini, N., 2005. On the nyström method for approximating a gram matrix for improved kernel-based learning. journal of machine learning research 6.

Friedman, J., Hastie, T., Tibshirani, R., 2010. A note on the group lasso and a sparse group lasso. arXiv preprint arXiv:1001.0736 .

Hiremath, A., Viswanathan, V.S., Bera, K., Shiradkar, R., Yuan, L., Armitage, K., Gilkeson, R., Ji, M., Fu, P., Gupta, A., Lu, C., Madabhushi, A., 2024. Deep learning reveals lung shape differences on baseline chest CT between mild and severe COVID-19: A multi-site retrospective study. Computers in Biology and Medicine 177, 108643. doi:https://doi.org/10.1016/j.compbiomed.2024.108643.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-Net: Convolutional networks for biomedical image segmentation, in: Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015, Springer. pp. 234–241.

Van Griethuysen, J.J., et al., 2017. Computational radiomics system to decode the radiographic phenotype. Cancer research 77, e104–e107.

Varghese, B.A., et al., 2021. Predicting clinical outcomes in COVID-19 using radiomics on chest radiographs. The British Journal of Radiology 94, 20210221.

Xue, Z., Yang, F., Rajaraman, S., Zamzmi, G., Antani, S., 2023. Cross dataset analysis of domain shift in CXR lung region detection. Diagnostics 13, 1068.