A SMOOTH LOW-RANK TIME BASIS FOR NEURAL ESTIMATION OF CUMULATIVE LONGITUDINAL MEANS

Anonymous authors
Paper under double-blind review

ABSTRACT

To characterize how longitudinal outcomes are driven by the covariate history, we consider modeling the conditional mean of the outcome at visit t, $\mu_t = \sum_{s \le t} g_s(\mathbf{X}_s)$, with nonlinear and unknown functions g_s of the concurrent covariate vector $\mathbf{X}_s$, $s \in \{1, \dots, t\}$. When g_s is additive elementwise across $\mathbf{X}_s$, the estimation is well studied in the setting of distributed-lag non-linear models (DLNM) at the standard spline rates. When the structural assumption of additivity on g_s is not imposed, little work addresses the sample sizes typical of cohort studies, hundreds to a few thousand subjects. In particular, DLNM cannot capture interactions, which costs prediction accuracy that no increase in sample size recovers. On the other hand, sequence models, such as recurrent networks, Transformers and state-space models, specify an unrestricted function of the covariate history and impose no structure, but must learn the functional form from the data; and the direct neural encoding of the cumulative mean model, with a separate network for each visit, has an estimation variance that grows with the number of visits. We propose SMOOTHCUM, a neural estimator that keeps the cumulative form of the model, shares one feature-extracting network across all visits, and expresses the visit-specific output layer through B-spline basis functions of time. For a sample-split training procedure, we establish an exact decomposition of the mean squared error into squared bias and variance, where the variance term grows with the number of basis functions and the width of the network's last hidden layer. Simulation experiments use this decomposition to quantify the bias-variance trade-off of SMOOTHCUM and to compare it with competing methods. The measured variance of SMOOTHCUM is linear in the number of basis functions, with slopes scaling correctly in sample size and signal-to-noise ratio; the cross-validated number of basis functions adapts to the smoothness of the true time-varying coefficient; the advantage over the estimator with a separate network per visit grows with the number of visits, by a factor of 1.8 to 3.2; and under irregular observation times, where a separate network per visit cannot be built, SMOOTHCUM outperforms all competing methods, including DLNM, by a factor of 1.4 to 1.8. An analysis of cohort data agrees with the bias-variance decomposition: the advantage of SMOOTHCUM appears at cohort sample sizes and disappears in cohorts of thousands of subjects, where the variance it removes is small. When the model class is unknown, combining DLNM and SMOOTHCUM by model averaging, with weights chosen by cross-validation over subject-level folds, was not worse than a single model in any setting we evaluated.

1 INTRODUCTION

We aim to train a cumulative mean model of the form (1) using a dataset containing many short time series, each from one subject. A subject i is measured at visits $t \in \{1, \dots, T\}$ with a d-element covariate vector $\mathbf{X}_{it} \in \mathbb{R}^d$ and a scalar outcome $Y_{it} \in \mathbb{R}$, and the time-course mean of the outcome is driven by the history,

$$\mu_{it} = \mathbb{E}\big(Y_{it} \mid \mathbf{X}_{i1}, \dots, \mathbf{X}_{it}\big) = \sum_{s=1}^{t} g_s(\mathbf{X}_{is}), \tag{1}$$

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

where each g_s is a d-dimensional nonlinear and unknown function. The primary model training task is to estimate μ from the longitudinally repeated measurements from n independent subjects, in the setting of a typical application with a short time series of T (*say*, $T \leq 25$) and the sample size n in the hundreds to low thousands.

The cumulative mean model plays an important role in predicting an expected future outcome given observed historical covariates, such as in infectious disease surveillance, ICU patient monitoring and other longitudinal cohort studies. These studies essentially collect time-course data to predict future outcomes required in the process of decision-making; for example, assessing the impact of cumulative-exposure on cardiometabolic burden and cognitive decline (Gottesman et al., 2017) in an adult cohort and a pediatric cohort (Sunyer et al., 2015), as well as the influence of diet choices on growth and development among adolescents (Victora et al., 2008). These cohort studies fix their sample sizes in advance, at hundreds to a few thousand subjects, by power analysis subject to study cost and expected attrition.

The supervised learning problem is partially solved, depending on the form assumed for g_s. When every g_s is additive across covariates, model (1) becomes the distributed-lag non-linear model (DLNM; Gasparrini et al., 2010), and spline-based estimation of these low-dimensional functions is efficient, with little room for improvement. When additivity is not imposed, no existing method is satisfactory at cohort sample sizes. DLNM cannot represent interactions among covariates, so it has an approximation error that no sample size removes. Sequence models such as the GRU, the causal Transformer and the TCN impose no structure on g_s and must estimate the full d-dimensional functions from data, which costs accuracy at these sample sizes. The direct neural encoding of (1), one sub-network per visit, is correctly specified but pays in variance: its output layer has one set of coefficients for every visit, and at cohort sample sizes these coefficients are dominated by variance. Smoothing the per-visit estimates afterwards (Fan & Zhang, 2000) does not recover the loss. We list the competing estimators, the structure each imposes on (1) and their errors in Appendix B (Table 3). The key challenge is therefore to keep the cumulative structure of (1), a sum of visit-specific contributions, when g_s is non-additive, without paying for the structure with one output layer per visit.

We propose SMOOTHCUM, a neural estimator of the cumulative mean model (1), and make three contributions. (i) *Non-additive cumulative means at cohort sample sizes.* We keep the cumulative structure of (1) but do not impose additivity, so we estimate non-additive g_s at cohort sample sizes, where DLNM is misspecified and unstructured sequence models are too variable. (ii) *A smooth low-rank representation of the time dimension.* We represent the time dimension of the output layer by a small number K of fixed smooth basis functions of time, in place of T separate per-visit output layers. The contribution of each visit is then an explicit smooth function of time with K coefficients, which a sequence model does not provide, and the complexity of the time dimension is controlled by the single integer K, chosen by cross-validation. We support this construction by theory. For a sample-split version of the estimator we derive an exact decomposition of the risk into squared bias and variance (§5), in which the variance term is proportional to K, so that replacing the T per-visit output layers by K basis coefficients divides the variance bound by T/K. (iii) *Time-invariant features shared across visits.* We share one feature-extracting network across all visits, so the features are estimated from all nT observations rather than from n per visit. Sharing assumes that all g_s lie in the span of one common set of features (§4). It is the larger of the two sources of variance reduction in our experiments (§6.2), and a multitask bound supports it (Appendix A.5).

We design our evaluation to test the bias-variance explanation of why SMOOTHCUM improves on the sub-network estimator, not only to rank the competing methods. In §6 we establish three claims.

1. *The gain is variance, not bias.* On a fixed evaluation design we estimate the squared bias and the variance of each estimator by Monte Carlo over 60 refits. The improvement of SMOOTHCUM over the sub-network estimator is a reduction in variance at comparable bias, as the risk decomposition suggests; the measured variance is linear in q with slopes that scale with n and the noise level as suggested, and sharing one network across visits adds a further factor of 1.7 to 3.1 (§6.2). The cross-validated K adapts to the smoothness of the time-varying coefficient and the advantage grows with T (Appendix C).
2. *When g_s is non-additive,* SMOOTHCUM *has the smallest error of every method compared*, including DLNM, by a factor of 1.9 on the comparison grid (§6.3) and by a factor of 1.4 to 1.8 under irregular observation times, where a per-visit output layer cannot be constructed

but a spline basis in continuous time can (§6.3). The margin is maintained against recent state-space and patch-transformer architectures (Appendix C).

3. *When g_s is additive, DLNM keeps a bias advantage* that no sample size removes (§6.3), and model averaging was not worse than a single model in any of 12 settings (§7); cohort data agree (§6.4).

These results give a rule for practice: when the target is non-additive or the observation times are irregular, use SMOOTHCUM; when the target is additive, use DLNM; when the model class is unknown, which is the usual case, combine the two by model averaging with weights chosen by cross-validation over subject-level folds, which in our experiments is never worse than a single model in any of 12 settings (§7). Two boundaries of the advantage are part of the result: it shrinks as inactive nuisance covariates are added, and reducing the covariate dimension restores it (Appendix C); and it disappears at large n, as any reduction in variance must (§6.4).

2 RELATED WORK

The most related statistical procedures impose a cumulative structure through a fixed functional form, or smooth after estimation. DLNM (Gasparrini et al., 2010) is the standard estimation procedure for cumulative exposure effects. The historical functional linear model (Malfait & Ramsay, 2003) is the special case of model (1) in which every g_s is linear. The two-step procedure of Fan & Zhang (2000) fits a separate regression at each visit and smooths the fitted curves across visits in a second stage. SMOOTHCUM differs from the standard two-step estimation procedure by embedding the time basis functions inside a neural network, so that we can perform both cross-visit smoothing and information pooling as part of estimation rather than a post-estimation operation.

Although the output-layer parameterization is not new, it is useful to formulate each visit-specific output layer as a neural version of the varying-coefficient model (Hastie & Tibshirani, 1993), while related fixed basis expansions in output layers appear in functional MLPs (Rossi & Conan-Guez, 2005) and N-BEATS (Oreshkin et al., 2020). We establish analytically, later confirm through simulation experiments, that the output-layer parameterization allows appropriate control of model complexity for the target (1) through an exact decomposition of the prediction risk into squared bias and variance, which helps tune the number of basis functions, K.

Sequence models impose no structure on the target (1). We compare against them in two groups. The first group consists of models for regularly spaced visits: the gated recurrent subject (GRU), a Transformer encoder with a causal mask, a temporal convolutional network (TCN) with dilated causal convolutions, a selective state-space model in the S4/S6 line (Gu et al., 2022; Gu & Dao, 2024), and a patch-tokenized transformer in the PatchTST line (Nie et al., 2023). Causal means that the prediction at visit t uses only the covariates observed at visits $s \leq t$; every method in this paper respects this restriction. The second group consists of continuous-time models for irregular observation times. There we compare against the ODE-RNN (Rubanova et al., 2019), a recurrent network whose hidden state evolves according to an ordinary differential equation between observations (§6.3). We do not compare against the neural controlled differential equation (Kidger et al., 2020), which drives the ODE-RNN hidden state with an interpolation of the covariates.

Semi-structured regression (Rügamer et al., 2024) combines a structured component and a deep component inside one jointly trained model, with an orthogonalization constraint that keeps the two components identifiable. Instead, in §7 we fit DLNM, SMOOTHCUM and a GRU separately, each as an estimator of the mean (1), and combine their predictions rather than their parameters. To estimate the combination weights we split the subjects into folds and use the predictions that each estimator makes on the subjects outside the fold it was fitted on. This is the standard model-averaging scheme, for which oracle-type guarantees exist (Wolpert, 1992; Breiman, 1996; van der Laan et al., 2007). Because the visits of one subject are correlated, the folds must keep all visits of a subject together (Hansen & Racine, 2012; Gao et al., 2016).

3 SETUP

We observe n independent subjects and evaluate every method as a predictor that uses only the covariate history. The data for subject i are $O_i = \{(Y_{it}, \mathbf{X}_{it}) : t \leq T\}$, $i = 1, \ldots, n$, with

conditional mean (1) and residuals $\varepsilon_{it} = Y_{it} - \mu_{it}$ that are correlated within a subject. Every method predicts Y_{it} from $\mathbf{X}_{i1}, \ldots, \mathbf{X}_{it}$ only. We call the true mean function (1) the target, and say that the target is additive when every g_s is additive across the elements of $\mathbf{X}_{is}$. Throughout, we distinguish two model classes, additive targets and non-additive targets.

DLNM fits $\hat{\mu}_{it} = \beta_0 + \sum_{j=1}^{d} \sum_{s \le t} f_j(X_{ijs}, t-s)$, where each f_j is a tensor-product basis of a spline in the covariate value and a polynomial in the scaled lag $t-s$, by ridge-penalized least squares on the pooled observations. It therefore imposes two restrictions: additivity across covariates, and indexing of time by the lag $t - s$ rather than by the visit s. We measure in §6.3 the cost of each restriction when it is violated and its benefit when it holds.

Model (1) is a marginal specification, modeling how the whole exposure history affects the outcome. A transitional model (Song, 2007; Li, 2004; Albert, 2000) conditions on the outcome's own past, $\mathbb{E}(Y_{it} \mid Y_{i,t-1}, \mathbf{X}_{it}) = \rho Y_{i,t-1} + h_t(\mathbf{X}_{it})$, modeling how the current exposure affects the outcome given the previous one. Because $Y_{i,t-1}$ carries the effect of all earlier exposures, h_t is an innovation effect, not the total cumulative effect. If the transitional model with a first-order autoregressive term holds, we extend SMOOTHCUM to the transitional class by adding an autoregressive term, as the marginal mean has the form (1) with the contribution of visit s weighted by ρ^{t-s}. We show in the simulation (Appendix E) that each class recovers its own estimand ($\hat{\rho} = 0.60$ at $\rho = 0.6$) and that inside the transitional class the additive restriction costs a factor of 3.6 when the innovation function is non-additive.

4 SMOOTHCUM

We replace the T separate networks of the sub-network estimator by one shared network and a fixed set of K smooth basis functions of time in the SMOOTHCUM model. Let $\boldsymbol{\phi}(\mathbf{x}) \in \mathbb{R}^H$ be the last hidden layer, of width H, of a feedforward network and $\tilde{\boldsymbol{\phi}}(\mathbf{x}) = (1, \boldsymbol{\phi}(\mathbf{x})^\top)^\top$ the same vector augmented with an intercept term. The sub-network estimator gives every visit its own network $\boldsymbol{\phi}_s$ and sets $g_s(\mathbf{x}) = \tilde{\boldsymbol{\phi}}_s(\mathbf{x})^\top \boldsymbol{\beta}_s$, so that its output layer has $T(H+1)$ coefficients. We instead set in SMOOTHCUM

$$g(s, \mathbf{x}) = \sum_{k=1}^{K} B_k(s)\, \tilde{\boldsymbol{\phi}}(\mathbf{x})^\top \boldsymbol{\beta}_k, \qquad \hat{\mu}_{it} = \sum_{s \le t} g(s, \mathbf{X}_{is}), \qquad q = K(H+1), \tag{2}$$

where $\{B_k\}_{k=1}^{K}$ is a fixed quadratic B-spline basis on $[1, T]$ with equally spaced knots, forming a partition of unity, that is, the B_k are nonnegative and sum to one at every s, and all network weights and the coefficient vectors $\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_K$ are estimated jointly by minimizing the squared-error loss. Here $g(s, \cdot)$ is the estimate of g_s; we write it without a hat to keep the notation light. We use quadratic splines because they are the lowest-degree splines with a continuous first derivative, so the estimated time-varying coefficient is smooth rather than piecewise linear with kinks at the knots, and because they permit the smallest nondegenerate smooth basis, K=3, on the short panels considered here; cubic splines would require $K \ge 4$.

We call the shared network the body and the output layer the head. Three properties follow from the construction. First, the shared network links the T per-visit regression problems: every observation contributes to the estimation of $\boldsymbol{\phi}$, and hence to the estimated contribution of every visit. Second, the smoothing takes place during estimation: because the B_k are fixed, the smoothness constraint on the time-varying coefficient is imposed while $\boldsymbol{\phi}$ is being estimated, unlike post-hoc smoothing of separately fitted curves (Fan & Zhang, 2000). Third, the complexity of the time-varying coefficient is governed by the single integer K: K=1 sums the contributions of all visits with equal weight, K=T recovers the per-visit output layers of the sub-network estimator, and K is chosen by cross-validation. Because (2) is defined at any real s, the estimator is also defined at irregular observation times, where a per-visit output layer cannot be constructed; we evaluate this setting directly in §6.3.

Sharing the network is also what makes smoothing across visits well defined. Smoothing the coefficient vectors $\boldsymbol{\beta}_s$ across visits presupposes that coordinate j of the last hidden layer has the same meaning at every visit. In contrast, with T separately trained networks, coordinate j need not have the same meaning at different visits. This is why Fan & Zhang (2000) smooth the fitted predictions rather than the coefficients. In our model (2), hidden unit j is the same function ϕ_j at every visit, with one set of weights evaluated at different inputs. The basis therefore smooths the coefficients

of one common set of features, and no matching is needed. Instead we assume that all g_s lie in the span of one common set of estimated features, that is, that the visit functions $g_1, \ldots, g_T$ are well approximated by combinations of a small number of common functions. This is the approximate low time-rank assumption of Appendix A, and we measure what happens when it fails (Appendix C.3).

An alternative is to supply the time basis as additional input coordinates, $g(s, \mathbf{x}) = f([\mathbf{x}, B(s)])$. We call this the variant with time as an input and report it throughout. On a regular visit grid its error is 0 to 21 percent lower than that of the product form (2) (Appendix C.4). Under irregular observation times the two are indistinguishable (§6.3). We treat the product form as the primary estimator because its output layer is linear in the coefficients, which is what makes the risk decomposition of §5 available; the claim of the paper concerns the structure shared by both variants.

5 THEORY

We analyze a sample-split version of the estimator, for which the risk has an exact decomposition into squared bias and variance. Split the subjects into two folds $\mathcal{I}_A$ and $\mathcal{I}_B$ of sizes n_A and n_B. Estimate the shared network $\boldsymbol{\phi}$ on fold A and hold it fixed. On fold B, subject i at visit t contributes the design row

$$Z_{it}^\top = \Big(\sum_{s \le t} B_1(s) \tilde{\boldsymbol{\phi}}(\mathbf{X}_{is})^\top, \; \ldots, \; \sum_{s \le t} B_K(s) \tilde{\boldsymbol{\phi}}(\mathbf{X}_{is})^\top \Big) \in \mathbb{R}^q,$$

so that $\hat{\mu}_{it} = Z_{it}^\top \boldsymbol{\beta}$ is linear in the stacked coefficient vector $\boldsymbol{\beta} = (\boldsymbol{\beta}_1^\top, \ldots, \boldsymbol{\beta}_K^\top)^\top$; the rows are stacked into $\mathbf{Z}_B$ with $N_B = n_B T$ rows and $\boldsymbol{\beta}$ is estimated by least squares. Let $P_K = \mathbf{Z}_B \mathbf{Z}_B^+$ be the projection onto the column space of $\mathbf{Z}_B$, r_K its rank, μ_B the stacked true means of fold B, Σ_B the block-diagonal covariance matrix of the fold-B residuals, and $\mathcal{F}_{A,X_B}$ the information in fold A and in the fold-B covariates, which fixes $\boldsymbol{\phi}$ and $\mathbf{Z}_B$. Identification of each g_s from the sequence of conditional means, which needs no assumption beyond integrability, is Theorem A1 in Appendix A.

Theorem 1 (Exact risk decomposition)**.** *Assume the subjects are independent with* $\mathbb{E}(\varepsilon_i \mid \mathbf{X}_i) = 0$ *and that* $\Sigma_i = \mathrm{Cov}(\varepsilon_i \mid \mathbf{X}_i)$ *satisfies* $\lambda_{\max}(\Sigma_i) \le \nu_T$ *a.s.* (A1)*. Then conditionally on* $\mathcal{F}_{A,X_B}$ *and for* any *conditional mean* μ_B*,*

$$\mathcal{R}_{\mathrm{in}} := \mathbb{E}\Big[\tfrac{\|\hat{\mu}_B - \mu_B\|_2^2}{N_B} \;\Big|\; \mathcal{F}_{A,X_B} \Big] = b_K^2 + \frac{\mathrm{tr}(P_K \Sigma_B)}{N_B} \le b_K^2 + \frac{\nu_T \, r_K}{n_B T} \le b_K^2 + \frac{\nu_T K(H{+}1)}{n_B T},$$

where $P_K = \mathbf{Z}_B \mathbf{Z}_B^+$*,* $r_K = \mathrm{rank}(\mathbf{Z}_B)$ *and* $b_K = \|(I - P_K)\mu_B\|_2 / \sqrt{N_B}$*. No structural assumption and no lower eigenvalue condition on* $\mathbf{Z}_B^\top \mathbf{Z}_B$ *enters.*

Corollary 1 (Coefficient count)**.** *The variance bound in Theorem 1 at basis size* K *is* K/T *times the bound at* $K{=}T$*, the per-visit output layer. The ratio* K/T *applies to the bound, and the effect of smoothing on total risk is decided by comparing this variance reduction with the bias term* b_K^2*.*

By Theorem 1, given the estimated network, the in-design risk $\mathcal{R}_{\mathrm{in}}$ is a squared approximation error b_K^2, which is the squared bias, plus a variance term bounded by $\nu_T K(H{+}1)/(n_B T)$. This risk decomposition holds for any conditional mean μ_B, so it does not require model (1) to be correct: misspecification enters only through b_K. The variance bound is linear in the number of output-layer coefficients $q = K(H{+}1)$, falls with the number of subjects, and rises with the noise level; replacing T visit-specific output layers by K basis coefficients divides it by T/K. Because ν_T grows with the within-subject residual dependence, $n_B T$ overstates the effective sample size, and strong dependence can cancel the factor $1/T$. The theorem describes the two-step construction, not the jointly trained estimator used in practice. Its role is to identify K as the tuning parameter and to explain the phenomenon observed in §6.2: the variance is linear in q, its slope falls with n and rises with the noise level, and the sub-network estimator, with $q = T(H{+}1)$, has a variance cost of order T/K relative to SMOOTHCUM.

We state and prove the remaining results in Appendix A. Under the low time-rank assumption of §4, a time-basis approximation assumption and a completeness assumption on the estimated features, we bound the approximation error of (2) by $T\Delta_{K,H,T}$, where $\Delta_{K,H,T}$ collects the three error levels (Theorem A2). This bounds the bias term b_K (Corollary A1), shows that the partition-of-unity property keeps it small for smooth time-varying coefficients, gives the number of basis functions that

minimizes the resulting upper bound (Corollary A2), and states what the coefficient count implies for realized variances (Corollary A3). These results cover the output layer only. For the shared network we adapt the multitask representation bound of Maurer et al. (2016), in which sharing reduces the representation term by a factor of $\sqrt{T}$, and we show that with a linear network sharing gives no reduction, so the benefit of sharing requires a nonlinear network (Proposition A1, Appendix A.5).

6 EXPERIMENTS

6.1 SIMULATION SETTINGS

All simulation studies share one data-generating process and one evaluation protocol. Covariates are Gaussian with $(\Sigma_X)_{jk} = 0.5^{|j-k|}$ and temporal dependence $\mathbf{X}_{it} = \rho_X \mathbf{X}_{i,t-1} + \sqrt{1-\rho_X^2}\,\boldsymbol{\xi}_{it}$, $\boldsymbol{\xi}_{it} \sim N(0, \Sigma_X)$ independent; the mean is $\mu_{it} = \sum_{s \le t} a_s g(\mathbf{X}_{is})$ with per-visit contribution

$$g_{\text{base}}(\mathbf{x}) = x_1^3 + x_2^2 + \sin x_3 + |x_4| + x_5 x_6 + 0.5e^{-x_7^2}, \qquad g_{\text{int}}(\mathbf{x}) = x_1 x_5^2 + \sin(x_2 x_7) + x_3 x_6 |x_4|, \tag{3}$$

where g_{base} is predominantly additive across covariates (its one interaction pairs adjacent, correlated covariates, and its additive share is $\pi_{\text{add}} = 0.98$, where π_{add} denotes the fraction of the variance of $g(\mathbf{X})$ captured by its best additive approximation under the covariate distribution) and g_{int} is not: its products pair covariates at lag ≥ 3 in Σ_X, so its additive share is only $\pi_{\text{add}} = 0.27$ and the covariate functions must be estimated jointly. Noise is scaled to $\text{SNR} = \text{Var}(\mu_{it})/\sigma_\varepsilon^2 = 2$ unless a study varies it, with within-subject residual correlation (AR(1) $\rho_\varepsilon = 0.5$ in the reference setting). Subjects are split $80/20$ into training and test identically across methods; all tuning takes place strictly inside the training subjects: the network width and learning rate are chosen by cross-validation at the subject level, the sequence models use early stopping on held-out training subjects, and K is chosen the same way for SMOOTHCUM; simulation scores are ISE against the true mean. A signal pattern is a pair of a covariate function g from (3) and a time-varying coefficient a_s. In simulation the true mean is known. We then score a method by the integrated squared error (ISE), the average of $(\hat{\mu}_{it} - \mu_{it})^2$ over the visits of independent test subjects. The ISE approximates the squared error integrated over the distribution of the covariate history. We call the expected ISE the risk. On real data the score is the prediction error on held-out subjects (§6.4). Four studies use this protocol: the comparison grid (5 signal patterns $\times$ 5 residual structures $\times$ $n \in \{200, 500\} \times \rho_X \in \{0, 0.3, 0.6\}$, 150 settings, 200 replicates), the decomposition study (§6.2; a fixed evaluation set of 1,000 subjects, 60 refits on fresh training samples, hyperparameters fixed at the values selected by cross-validation, so that q is exact), the smoothness sweep (Appendix C.3), and the scale grid (Appendix C.4; 40 replicates). Full designs and tables are in Appendix C.

6.2 BIAS-VARIANCE DECOMPOSITION

We first check, as Theorem 1 suggests, whether SMOOTHCUM improves on the sub-network estimator by reducing variance, not bias. The decomposition study draws one evaluation set of 1,000 subjects per setting with a fixed seed, refits every estimator on 60 fresh training samples, and computes the squared bias and the variance of the fitted mean pointwise across the refits; their sum equals the ISE. Hyperparameters are fixed at the values selected by cross-validation, so that $q = K(H+1)$ is exact. At T=5 and n=500, SMOOTHCUM with K=3 has squared bias 7.27 and variance 5.23, while the sub-network estimator has squared bias 10.13 and variance 14.58 (Figure 1b). The difference in ISE is almost entirely a difference in variance. At a poorly tuned setting (width 8, learning rate 10^{-3}) both estimators are dominated by bias and the difference in variance is not visible.

The variance behaves as the theorem suggests, in size as well as in direction. Within SMOOTHCUM, the variance is linear in q across $K = 1, \ldots, 5$ (Figure 1a; over the eight settings the largest deviation of any point from its fitted line is 8%). The slope falls with the sample size, 0.063, 0.024 and 0.014 at $n = 200$, 500 and 2000, and rises with the noise level, 0.093, 0.024 and 0.014 at SNR 0.5, 2 and 8 (Table 5, Appendix C). Linearity in q is immediate for least squares on a fixed design; the finding is that the jointly trained estimator follows the same law, slopes included.

Sharing the network is the larger of the two sources of variance reduction. At K=T the basis consists of the visit indicators, so SMOOTHCUM and the sub-network estimator have the same number of output-layer coefficients and differ only in sharing the network. The variance ratio there is between

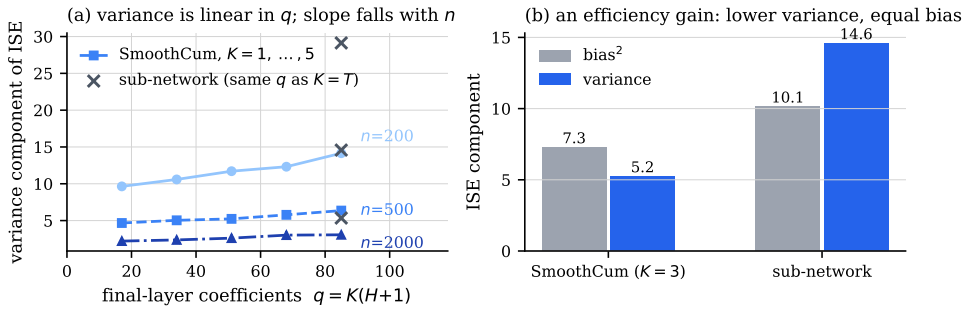

Figure 1: The bias-variance decomposition, measured on a fixed evaluation design (constant signal, T=5, SNR 2, 60 refits). *(a)* Variance component of ISE against the output-layer coefficient count $q = K(H+1)$ for $K = 1, \ldots, 5$: linear within SmoothCum at every n, with slope falling as n grows; gray crosses are the sub-network estimator, whose q equals SmoothCum's at K=T, so the vertical gap at matched q is the effect of sharing the network. *(b)* Bias^2/variance split at the reference setting: the two estimators differ almost entirely in variance.

1.74 and 3.14 (gray crosses in Figure 1a). Over the comparison grid the shared network accounts for 26% of the gap to the sub-network estimator and the basis for 12% (Appendix C).

The same holds on the non-additive target. For g_{int} at T=5 and n=500, the variance is again linear in q (slope 0.013), the variance ratio at equal capacity is 2.1, and the improvement over the sub-network estimator (4.05 against 9.67) is again a reduction in variance. The decomposition also shows why DLNM fails on this target: its squared bias is 10.9 against a variance of 1.1, so its error is bias, which no sample size reduces.

Two further implications of the decomposition are confirmed in Appendix C. The cross-validated K adapts to the smoothness of the true time-varying coefficient: on a sinusoidal coefficient of increasing frequency the selected K rises from 3 to T, and when the coefficient is less smooth than the basis can follow the estimator reduces to the per-visit output layers rather than falling behind them (Appendix C.3). The advantage over the sub-network estimator grows with T, from a ratio of 1.77 ± 0.09 at T=5 to 3.24 ± 0.12 at T=24 on non-additive targets, and is maintained against a selective state-space model and a patch transformer in 13 of 14 settings (Appendix C.4).

6.3 Comparison across methods

The comparison grid crosses the two covariate functions of (3) with five time-varying coefficients, five residual structures, two sample sizes and three levels of covariate dependence: 150 settings with 200 replicates each. On the non-additive target g_{int}, SmoothCum has the smallest error of all methods compared: ISE 8.44 against 16.13 for DLNM, a factor of 1.9, with the sequence models at 14.0 to 14.8 and the sub-network estimator at 13.3 (Table 1). Averaged over all 150 settings (Table 2), it is the best of the estimators that do not impose additivity: its ISE of 15.99 is 33% below the sub-network estimator (23.86), 38 to 52% below the TCN (25.72), the GRU (32.42) and the Transformer (33.16), and below the two-step estimator (25.08) and the gradient boosting machine (GBM, 19.92). Methods that use only the current covariates are far behind (46 to 50). All neural methods share one training and tuning step of about 40 seconds per replicate on a CPU, so their costs are alike, and DLNM, GBM and GEE each fit in about a second or less (Appendix G).

The margins over the sequence models reflect the structure imposed, not the training effort. The GRU, the Transformer and the TCN share the tuning grids of SmoothCum and use early stopping on held-out training subjects, so each is at its own validation optimum; their deficit is the cost of learning the cumulative form from data. DLNM leads the average over all settings (11.16) because four of the five time-varying coefficients are paired with the predominantly additive g_{base}; we therefore report the two model classes separately in Table 1. We add two supporting results in Appendix C: the margin over the sequence models holds against a selective state-space model and a patch transformer in 13 of 14 settings of a larger grid with T up to 24, d up to 50 and n up to 2000, and inactive nuisance covariates erode the margin, which dimension reduction restores.

Table 1: The split by model class (test ISE, mean over settings of the comparison grid). DLNM has the smaller error for additive targets and SMOOTHCUM for non-additive ones; neither the sub-network estimator nor the GRU wins in either class. Per-pattern detail: Table 4, Appendix C.

Method	additive targets (g_{base}, π_{add}=0.98)	non-additive (g_{int}, π_{add}=0.27)
DLNM	**9.93**	16.13
SMOOTHCUM	17.87	**8.44**
Subnet-Cum	26.49	13.32
GRU	37.02	14.04

Table 2: Comparison grid: 150 settings, 200 replicates each, mean over settings. ISE is against the true mean (1); per-setting results are provided as CSV files in the code repository. Subnet-Cum is the sub-network estimator, TwoStep the two-step estimator of Fan & Zhang (2000), FuncLinear the historical functional linear model, GEE a generalized estimating equation fit on the covariate history (hist.) or on the current covariates only (current), and NN (current) a feedforward network on current covariates.

Method	MSE	ISE	Method	MSE	ISE
SMOOTHCUM (ours)	46.90	**15.99**	GRU	63.35	32.42
DLNM	42.09	11.16	FuncLinear	63.94	33.03
GBM	50.85	19.92	Transformer	64.10	33.16
Subnet-Cum	54.80	23.86	GEE (hist.)	65.42	34.47
TwoStep	56.04	25.08	NN (current)	77.33	46.32
TCN	56.63	25.72	GEE (current)	80.48	49.50

Irregular observation times. Cohorts observe subjects at subject-specific times. A per-visit output layer requires a common visit grid and cannot be constructed there, whereas (2) is defined at any real s, so SMOOTHCUM applies directly. We simulate this setting: each of n=500 subjects is observed at its own T=8 sorted $\text{Uniform}(0,1)$ times $t_{i1} < \cdots < t_{iT}$, the mean at t_{ij} is $\sum_{l \le j} a(t_{il}) g(\mathbf{X}_{il})$ with $a(t) = 1 + 0.8\sin(2\pi f t)$ at two smoothness levels $f \in \{1, 2\}$ and $g \in \{g_{\text{base}}, g_{\text{int}}\}$, and the basis is evaluated at the true times (40 replicates). The competing methods are the standard methods for irregularly sampled series: a GRU that receives the observation time and the time gap as inputs, and an ODE-RNN (Rubanova et al., 2019), whose hidden state follows an estimated ordinary differential equation between observations; DLNM uses a continuous lag basis.

On g_{int} at f=1, SMOOTHCUM attains ISE 0.283 ± 0.014 against 0.400 ± 0.021 for the GRU (a factor of 1.41), 0.414 ± 0.020 for the ODE-RNN (1.46) and 0.519 ± 0.021 for DLNM (1.84, with the smaller error in all 40 replicates; all Wilcoxon $p < 10^{-4}$). The continuous lag basis does not repair the additivity restriction of DLNM. At f=2 the factors are 1.24 and 1.51. The variant with time as an input, ahead by 0 to 21% on regular grids, ties the product form (ratios 0.998 and 0.978, $p \ge 0.64$). On the additive g_{base}, DLNM has the smaller error at both smoothness levels, for the reason given next: with an additive target the DLNM restriction is correct, and its advantage is bias.

The additive case. We split the comparison grid by model class in Table 1. On the predominantly additive g_{base}, DLNM has a smaller error than SMOOTHCUM by a factor of 1.7 to 2.3 in three of the four settings. In the fourth setting, whose time-varying coefficient $a_s = s/T$ depends on absolute time and so violates DLNM's lag indexing, the two tie (6.87 against 6.94). On g_{int}, DLNM has a larger error by a factor of 1.9 at both sample sizes. The relative performance is determined by which restriction the target violates, additivity ($\pi_{\text{add}} = 0.98$ against 0.27) or lag indexing. The ordering is the same at both sample sizes of the grid. Repeating the decomposition of §6.2, we find that the advantage of DLNM is bias: on additive targets its squared bias is 3.8 against 7.3 for SMOOTHCUM, and no reduction in variance closes that gap. This sets the boundary of the method: SMOOTHCUM should replace DLNM when the target is non-additive or the observation times are irregular, and not otherwise.

6.4 APPLICATION: DIET HISTORY AND CHILD GROWTH

We apply the proposed method to the ELEMENT birth cohort. Its sample size is small enough that the variance term of Theorem 1 is not negligible. In ELEMENT, children are followed with diet recalls and anthropometry (n=665 children, T=5 visits at ages 12–36 months, d=17 nutrients), and we predict the WHO weight-for-length z-score at each visit from the cumulative history of total-energy-adjusted nutrient intake under model (1). The models are evaluated by the mean squared prediction error on held-out subjects (test MSE) over repeated subject-level $80/20$ splits, with all tuning inside the training subjects.

Among the estimators that do not impose additivity, SMOOTHCUM has the smallest error: the best sequence model (the GRU) has 1.37 times its test MSE and the sub-network estimator 1.16 times, the same ordering as in the simulations at this sample size. DLNM has 8% lower test MSE, consistent with predominantly additive nutrient effects; this is the bias advantage of §6.3 on real data. The cohort's second exposure panel, d=40 food groups, reproduces the erosion with covariate dimension and its repair by principal components. In two public cohorts with thousands of subjects the advantage vanishes, as the decomposition suggests. On HRS cognition (n=2510) the error ratios of all competing methods to SMOOTHCUM are 0.99 to 1.01. On ECLS-K reading (n=6318) the sub-network estimator and the best sequence model are ahead, with ratios 0.71 and 0.70 (Appendix F).

7 MODEL AVERAGING WHEN THE MODEL CLASS IS UNKNOWN

Choosing between DLNM and SMOOTHCUM is costly when the guess about the model class is wrong: SMOOTHCUM loses by a factor of 1.7 to 2.3 on additive targets (§6.3) and DLNM by a factor of 1.9 on non-additive ones (§6.3). When the class of a real target is unknown, we therefore recommend model averaging (Wolpert, 1992; Breiman, 1996; van der Laan et al., 2007): a weighted average of the predictions of DLNM, SMOOTHCUM and a GRU, with non-negative weights estimated from out-of-fold predictions on folds formed at the subject level. It needs only the refits that cross-validated model selection already performs, and it has no tuning constant. In a design that crosses the two covariate functions of (3) with three time-varying coefficients at $n \in \{200, 500\}$ (12 settings, 100 replicates, V=4 subject-level folds), the average has a lower mean ISE than the single model selected by cross-validation in all 12 settings (by 6.7% on average, significant in 8 by a paired t-test over replicates). On the six additive settings it matches the selected single model, DLNM, so carrying SMOOTHCUM costs nothing there; on the six non-additive settings the roles are reversed (Appendix D). The gain comes from including SMOOTHCUM. The average of DLNM and the GRU alone has a higher mean ISE than the full pool in all 12 settings: by 1 to 8% on the additive settings with a constant or decaying coefficient, by 21 to 48% on the additive settings with an increasing coefficient, and by a factor of 1.3 to 2.0 on the non-additive settings (Table 14, Appendix D).

8 LIMITATIONS AND CONCLUSION

Three limitations bound the results. First, the risk decomposition covers a sample-split construction, not the jointly trained estimator; its implications are what the experiments test. For the shared network, the larger source of variance reduction, we have only a bound adapted from multitask learning (Appendix A.5). Second, the advantage is a reduction in variance: it disappears at large n, is bounded by the bias advantage of DLNM on additive targets, and shrinks with nuisance covariates unless their dimension is reduced. Third, no cohort in our set has a non-additive target, so the real-data evidence covers the additive class only. Within its model class and sample-size range, a shared network with a cross-validated rank-K smooth time basis in the output layer is the best estimator we measured: it removes variance with comparable bias in the settings evaluated.

REPRODUCIBILITY STATEMENT

All simulation studies are fully specified: the data-generating process, signal functions and noise scaling are in §6.1 and equation (3); grids, tuning protocols, early-stopping settings and the two recent architectures' hyperparameters are in Appendices C and G; the decomposition study's fixed-seed evaluation design and operating point are stated in §6.2 and Appendix C. Every tuning choice, including K, is made inside training subjects. The cohort analysis reports its window,

panels, standardization and split protocol in Appendix F; ELEMENT data are available through the cohort's access procedure. Code for all estimators, the simulation grids and the figures is available at `https://anonymous.4open.science/r/SmoothCum_Publish-28C3/`, together with the summary files from which every table and figure is produced and, as CSV files, the per-setting results for all 150 settings of the comparison grid.

ETHICS STATEMENT

This work analyzes secondary, de-identified data from three existing cohorts. The ELEMENT birth cohort obtained informed consent from participants and approval from the institutional review boards of the participating institutions; we accessed the data under the cohort's data-use procedure and report no individual-level records. The HRS data are the public RAND HRS Longitudinal File, available after registration, and the ECLS-K:2011 data are the public-use file; both are released by their custodians in de-identified form. No new data were collected for this paper, and no individual-level data are released with it. The simulation studies use synthetic data generated from the process stated in §6.1. The estimator is a tool for estimating cumulative exposure effects in longitudinal studies, and we foresee no harmful application specific to it. The authors declare no competing interests.

AI USE STATEMENT

In this work we used a large-language-model (LLM) assistant in four ways, matching the disclosure made on the submission form. (i) *Research ideation and execution*: the assistant was used to discuss the simulation design, to assist in implementing the estimators and the experimental pipeline, and to interpret the results. (ii) *Proving mathematical claims*: the mathematical claims and their proofs were formulated and written by the authors; the assistant was used only to check steps in the proofs of Appendix A, and every statement and proof was subsequently re-derived and verified by the authors. (iii) *Drafting*: the assistant was used to summarize the experimental results and to produce initial drafts of parts of the manuscript. (iv) *Writing polish*: the assistant was used to edit the manuscript for readability and consistency. Among the tasks with recommended disclosure, the assistant was also used to refactor the experiment code and to generate figures and tables from the stored results. We did not use generative AI to generate any data reported in this paper, to formulate the hypotheses, to translate, to clean or reformat a data set, or for qualitative data analysis. We reviewed all AI-assisted content. Proofs were checked independently of the assistant, AI-assisted code was tested and its outputs were verified against the stored per-replicate result files, and every numerical claim in the text was checked against the computed results. We take responsibility for the final content of this work, including text, claims and artifacts produced with the aid of generative AI.

REFERENCES

Paul S Albert. A transitional model for longitudinal binary data subject to nonignorable missing data. *Biometrics*, 56(2):602–608, 2000.

Shaojie Bai, J Zico Kolter, and Vladlen Koltun. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271*, 2018.

Leo Breiman. Stacked regressions. *Machine Learning*, 24(1):49–64, 1996.

Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1724–1734, 2014.

Jianqing Fan and Jin-Ting Zhang. Two-step estimation of functional linear models with applications to longitudinal data. *Journal of the Royal Statistical Society: Series B*, 62(2):303–322, 2000.

Jerome H Friedman. Greedy function approximation: a gradient boosting machine. *The Annals of Statistics*, 29(5):1189–1232, 2001.

Yan Gao, Xinyu Zhang, Shouyang Wang, and Guohua Zou. Model averaging based on leave-subject-out cross-validation. *Journal of Econometrics*, 192(1):139–151, 2016.

Antonio Gasparrini, Ben Armstrong, and Michael G Kenward. Distributed lag non-linear models. *Statistics in Medicine*, 29(21):2224–2234, 2010.

Rebecca F Gottesman, Marilyn S Albert, Alvaro Alonso, Laura H Coker, Josef Coresh, Sonia M Davis, Jennifer A Deal, Guy M McKhann, Thomas H Mosley, A Richey Sharrett, Andrea L C Schneider, B Gwen Windham, Lisa M Wruck, and David S Knopman. Associations between midlife vascular risk factors and 25-year incident dementia in the Atherosclerosis Risk in Communities (ARIC) cohort. *JAMA Neurology*, 74(10):1246–1254, 2017.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. In *Conference on Language Modeling (COLM)*, 2024.

Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. In *International Conference on Learning Representations*, 2022.

Bruce E Hansen and Jeffrey S Racine. Jackknife model averaging. *Journal of Econometrics*, 167 (1):38–46, 2012.

Trevor Hastie and Robert Tibshirani. Varying-coefficient models. *Journal of the Royal Statistical Society: Series B*, 55(4):757–796, 1993.

Patrick Kidger, James Morrill, James Foster, and Terry Lyons. Neural controlled differential equations for irregular time series. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

Wai Keung Li. *Diagnostic Checks in Time Series*. Chapman & Hall/CRC, 2004.

Nicole Malfait and James O Ramsay. The historical functional linear model. *Canadian Journal of Statistics*, 31(2):115–128, 2003.

Andreas Maurer, Massimiliano Pontil, and Bernardino Romera-Paredes. The benefit of multitask representation learning. *Journal of Machine Learning Research*, 17(81):1–32, 2016.

Yuqi Nie, Nam H. Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In *International Conference on Learning Representations*, 2023.

Boris N Oreshkin, Dmitri Carpov, Nicolas Chapados, and Yoshua Bengio. N-BEATS: neural basis expansion analysis for interpretable time series forecasting. In *International Conference on Learning Representations*, 2020.

Wei Perng, Martha M Téllez-Rojo, Deborah J Watkins, et al. Early life exposure in Mexico to ENvironmental Toxicants (ELEMENT) project. *BMJ Open*, 9(8):e030427, 2019.

RAND Center for the Study of Aging. RAND HRS longitudinal file 2022 (V1). Santa Monica, CA: RAND, 2025.

Fabrice Rossi and Brieuc Conan-Guez. Functional multi-layer perceptron: a non-linear tool for functional data analysis. *Neural Networks*, 18(1):45–60, 2005.

Yulia Rubanova, Ricky T Q Chen, and David Duvenaud. Latent ODEs for irregularly-sampled time series. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

David Rügamer, Chris Kolb, and Nadja Klein. Semi-structured distributional regression. *The American Statistician*, 78(1):88–99, 2024.

Peter X-K Song. *Correlated Data Analysis: Modeling, Analytics, and Applications*. Springer Series in Statistics. Springer, 2007.

Amanda Sonnega, Jessica D Faul, Mary Beth Ofstedal, Kenneth M Langa, John W R Phillips, and David R Weir. Cohort profile: the Health and Retirement Study (HRS). *International Journal of Epidemiology*, 43(2):576–585, 2014.

Jordi Sunyer, Mikel Esnaola, Mar Alvarez-Pedrerol, et al. Association between traffic-related air pollution in schools and cognitive development in primary school children: a prospective cohort study. *PLoS Medicine*, 12(3):e1001792, 2015.

Karen Tourangeau, Christine Nord, Thanh Lê, Alberto G Sorongon, and Michelle Najarian. Early childhood longitudinal study, kindergarten class of 1998–99 (ECLS-K): Combined user's manual for the ECLS-K eighth-grade and K–8 full sample data files. Technical Report NCES 2009-004, National Center for Education Statistics, 2009.

Mark J van der Laan, Eric C Polley, and Alan E Hubbard. Super learner. *Statistical Applications in Genetics and Molecular Biology*, 6(1):Article 25, 2007.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Cesar G Victora, Linda Adair, Caroline Fall, Pedro C Hallal, Reynaldo Martorell, Linda Richter, and Harshpal Singh Sachdev. Maternal and child undernutrition: consequences for adult health and human capital. *The Lancet*, 371(9609):340–357, 2008.

David H Wolpert. Stacked generalization. *Neural Networks*, 5(2):241–259, 1992.

A THEORY: ASSUMPTIONS AND COMPLETE PROOFS

We give here the full assumption set and complete proofs of every theoretical result: identification, the exact risk decomposition (Theorem 1 of the main text, restated here), the approximation decomposition and the bound-optimal basis size.

The results isolate what can be proved for a *frozen learned representation*. The deployed estimator trains all parameters jointly; the sample split and the frozen body below are analysis devices that make an exact identity available, not features of the implementation. We first show that the additive-over-time target is identified without conditioning on the learned features (Theorem A1). We then bound a conditional *in-design fitted-mean risk* whose exact bias–variance identity holds for *any* conditional mean under a single noise assumption (Theorem A3); low time rank enters only as a sufficient bound on the bias. No result asserts an exact K/T ratio for realized variances, or that stochastic-gradient training attains the representation event it conditions on.

A.1 SETTING AND NOTATION

Basis on the visit grid. Assume $T \geq 2$ and write $u_s = (s-1)/(T-1)$. For each permitted K let $B^{K,T}(s) = (B_1^{K,T}(s), \ldots, B_K^{K,T}(s))^\top$ be the time basis evaluated at visit s. For $K \geq 3$ the implemented smooth candidates use a nonnegative partition-of-unity quadratic B-spline basis. The remaining cases are handled separately and are *not* called quadratic splines: $K = 1$ is the constant basis, the implemented $K = 2$ candidate is piecewise linear, and $K = T$ may be taken to be the indicator basis $B_k^{T,T}(s) = \mathbf{1}\{k = s\}$. Throughout, "smooth candidate" refers only to $3 \leq K < T$.

Data and notation. Subject i contributes $O_i = \{(Y_{it}, \mathbf{X}_{it}) : t \leq T\}$, where $\mathbf{X}_{it} \in \mathcal{X} \subseteq \mathbb{R}^d$ is the covariate vector and $Y_{it} \in \mathbb{R}$ the outcome at visit t. Write $Y_i = (Y_{i1}, \ldots, Y_{iT})^\top$, $\mu_i = (\mu_{i1}, \ldots, \mu_{iT})^\top$ with $\mu_{it} = \mathbb{E}(Y_{it} \mid \mathbf{X}_{i1}, \ldots, \mathbf{X}_{it}) = \sum_{s \leq t} g_s(\mathbf{X}_{is})$, $\varepsilon_i = Y_i - \mu_i$, and $\mathbf{X}_i = (\mathbf{X}_{i1}, \ldots, \mathbf{X}_{iT})$. For a set of subjects $\mathcal{I}$, the stacked vectors $Y_\mathcal{I}$, $\mu_\mathcal{I}$ and $\varepsilon_\mathcal{I}$ list the subject vectors one after another. $P_{\mathbf{X}_{is}}$ denotes the distribution of $\mathbf{X}_{is}$; $\|\cdot\|_2$ and $\|\cdot\|_\infty$ are the Euclidean and supremum norms; I is the identity matrix; $\preceq$ is the Loewner order on symmetric matrices; $\mathrm{col}(\cdot)$ is the column space; $\lambda_{\max}$ is the largest eigenvalue.

Sample-split frozen-body estimator. Split the independent subjects into folds $\mathcal{I}_A, \mathcal{I}_B$ of sizes n_A, n_B, and write Y_B, μ_B, ε_B for the fold-B stacked vectors. A body is a feedforward network $\boldsymbol{\phi}_\eta : \mathcal{X} \to \mathbb{R}^H$ with weights η and a last hidden layer of width H; $\tilde{\boldsymbol{\phi}}_\eta(\mathbf{x}) = (1, \boldsymbol{\phi}_\eta(\mathbf{x})^\top)^\top$ augments it with an intercept. The theoretical estimator is: (1) estimate the weights on fold A, giving a measurable shared body $\hat{\boldsymbol{\phi}} = \boldsymbol{\phi}_{\hat{\eta}_A}$ with $\tilde{\hat{\boldsymbol{\phi}}} = (1, \hat{\boldsymbol{\phi}}^\top)^\top$; (2) freeze it and form the fold-B cumulative-design row

$$Z_{it}^\top = \Big(\sum_{s \leq t} B_1^{K,T}(s) \tilde{\hat{\boldsymbol{\phi}}}(\mathbf{X}_{is})^\top, \ldots, \sum_{s \leq t} B_K^{K,T}(s) \tilde{\hat{\boldsymbol{\phi}}}(\mathbf{X}_{is})^\top \Big) \in \mathbb{R}^q, \qquad q = K(H+1);$$

(3) stack these into $\mathbf{Z}_B \in \mathbb{R}^{N_B \times q}$, $N_B = n_B T$, and set $\hat{\boldsymbol{\beta}}_B = \mathbf{Z}_B^+ Y_B$, $\hat{\mu}_B = \mathbf{Z}_B \hat{\boldsymbol{\beta}}_B$, where $^+$ is the Moore–Penrose inverse (unique and Borel measurable, so $\hat{\mu}_B$ is a measurable function of the data). Define $\mathcal{F}_{A,X_B} := \sigma\{O_i : i \in \mathcal{I}_A\} \vee \sigma\{\mathbf{X}_i : i \in \mathcal{I}_B\}$, which fixes $\hat{\boldsymbol{\phi}}$, $\mathbf{Z}_B$, μ_B and $\Sigma_B := \mathrm{blockdiag}(\Sigma_i : i \in \mathcal{I}_B)$, with Σ_i the conditional residual covariance of **(A1)**. The risk analyzed is

$$\mathcal{R}_{\mathrm{in}}(\hat{\mu}) := \mathbb{E}\Big[\tfrac{\|\hat{\mu}_B - \mu_B\|_2^2}{N_B} \;\Big|\; \mathcal{F}_{A,X_B} \Big]. \tag{4}$$

A single split is used; the symmetric cross-fit averages two such splits and every bound applies fold by fold.

A.2 ASSUMPTIONS

(A1) *Subjects and noise.* The subjects are independent, $\mathbb{E}(\varepsilon_i \mid \mathbf{X}_i) = 0$, and $\Sigma_i := \mathrm{Cov}(\varepsilon_i \mid \mathbf{X}_i)$ satisfies $\lambda_{\max}(\Sigma_i) \leq \nu_T$ almost surely. Write $\nu_T = \sigma_\varepsilon^2 \bar{\kappa}_T$, with σ_ε^2 the residual variance and $\bar{\kappa}_T \geq 1$ a factor measuring within-subject dependence; under strong equicorrelation $\bar{\kappa}_T \asymp T$, which cancels the $1/T$ in the variance term below.

(A2) *Approximate low time rank.* For a fixed K_0 and residual level $\zeta_{K_0} \geq 0$,

$$g_s(\mathbf{x}) = \sum_{r=1}^{K_0} c_r(u_s) h_r(\mathbf{x}) + \delta_s(\mathbf{x}), \qquad \|h_r\|_\infty \leq 1, \quad \|c_r\|_\infty \leq L, \quad \sup_{s \leq T} \|\delta_s\|_\infty \leq \zeta_{K_0}.$$

Here $c_r : [0,1] \to \mathbb{R}$ are time-coefficient functions evaluated at u_s, $h_r : \mathcal{X} \to \mathbb{R}$ are common covariate functions, $L > 0$ is a constant and δ_s is the residual. Only the rank K_0, the spanned subspace and ζ_{K_0} are invariant; the normalization $\|h_r\|_\infty \leq 1$ is without loss for nonzero bounded factors, whose scale is absorbed into c_r.

(A3) *Time-basis approximation.* For each r and permitted K there is $\theta_r^{K,T} \in \mathbb{R}^K$ with

$$\max_{s \leq T} \left| c_r(u_s) - B^{K,T}(s)^\top \theta_r^{K,T} \right| \leq a_{K,T}. \tag{5}$$

This is a direct finite-grid premise for the basis actually used. For $K = 1$ and the implemented $K = 2$ candidate, $a_{K,T}$ is their actual best grid error; no quadratic-spline rate is asserted there. For the indicator endpoint $a_{T,T} = 0$ exactly.

(A4) *Realized representation completeness.* On an $\mathcal{F}_{A,X_B}$-measurable event $\mathcal{E}_{A,H}$, for every $r \leq K_0$ there is $\gamma_r \in \mathbb{R}^{H+1}$ with

$$\sup_{\mathbf{x} \in \mathcal{X}} \left| h_r(\mathbf{x}) - \tilde{\hat{\boldsymbol{\phi}}}(\mathbf{x})^\top \gamma_r \right| \leq \epsilon_H, \tag{6}$$

so $v_r(\mathbf{x}) := \tilde{\hat{\boldsymbol{\phi}}}(\mathbf{x})^\top \gamma_r$ obeys $\|v_r\|_\infty \leq 1 + \epsilon_H$. This is a *completeness* (expressiveness) condition on the frozen span, not an identifiability condition on $\hat{\boldsymbol{\phi}}$, which is unidentified (see the paragraph "Non-identifiability of the frozen body is harmless" below) and never used as an identified object. No claim is made that stochastic-gradient training attains $\mathcal{E}_{A,H}$.

A.3 Statements

Theorem A1 (Identification by differencing)**.** *Assume the additive target $\mu_{it} = \sum_{s \leq t} g_s(\mathbf{X}_{is})$ with integrable g_s. For every $t \geq 2$, on the support of $(\mathbf{X}_{i1}, \ldots, \mathbf{X}_{it})$,*

$$g_t(\mathbf{X}_{it}) = \mu_{it} - \mu_{i,t-1} \quad \textit{almost surely}, \qquad g_1(\mathbf{X}_{i1}) = \mu_{i1}, \tag{7}$$

and the increment is a function of $\mathbf{X}_{it}$ alone. Hence every g_s is identified from the sequence of conditional means up to $P_{\mathbf{X}_{is}}$-almost-everywhere equivalence, with no centering constraint. If each g_s is continuous, identification is pointwise on $\operatorname{supp}(P_{\mathbf{X}_{is}})$.

Increment regressions require sequential exogeneity. Theorem A1 differences *conditional means*. The corresponding statement about the observable increment, $\mathbb{E}[Y_{it} - Y_{i,t-1} \mid \mathbf{X}_{i1}, \ldots, \mathbf{X}_{it}] = g_t(\mathbf{X}_{it})$, requires in addition $\mathbb{E}[\varepsilon_{i,t-1} \mid \mathbf{X}_{i1}, \ldots, \mathbf{X}_{it}] = 0$: since the additive target only defines $\mu_{i,t-1}$ as the mean given $(\mathbf{X}_{i1}, \ldots, \mathbf{X}_{i,t-1})$, replacing $\mathbb{E}[Y_{i,t-1} \mid \mathbf{X}_{i1}, \ldots, \mathbf{X}_{it}]$ by $\mu_{i,t-1}$ needs the current covariate to carry no information about the previous outcome shock (sequential exogeneity; no outcome-to-covariate feedback). The simulation designs in this paper satisfy it by construction, covariates evolve as an exogenous AR(1) independent of the residuals, and no estimator or evaluation in the paper uses the increment-regression form.

Non-identifiability of the frozen body is harmless. $\hat{\boldsymbol{\phi}}$ and $\hat{\boldsymbol{\beta}}_B$ are *not* identified: any invertible reparametrization $\hat{\boldsymbol{\phi}} \mapsto A\hat{\boldsymbol{\phi}}$, $\boldsymbol{\beta}_k \mapsto A^{-\top}\boldsymbol{\beta}_k$ leaves every fitted function unchanged, and $\hat{\boldsymbol{\beta}}_B$ is further non-unique whenever $\mathbf{Z}_B$ is rank-deficient. No risk result uses either as an identified object: all depend on the data only through $\operatorname{col}(\mathbf{Z}_B)$ and $\hat{\mu}_B = P_K Y_B$, both invariant to the reparametrization and defined with no rank condition. What the theory needs is **(A4)**-completeness of the span, a property of the realized features.

Theorem A2 (Approximation-error decomposition)**.** *Under **(A2)–(A4)**, on $\mathcal{E}_{A,H}$ there are coefficients $\boldsymbol{\beta}_1^\dagger, \ldots, \boldsymbol{\beta}_K^\dagger$ with*

$$\max_{s \leq T} \sup_{\mathbf{x} \in \mathcal{X}} \left| g_s(\mathbf{x}) - \sum_{k=1}^{K} B_k^{K,T}(s) \tilde{\hat{\boldsymbol{\phi}}}(\mathbf{x})^\top \boldsymbol{\beta}_k^\dagger \right| \leq \Delta_{K,H,T} := K_0\{L\epsilon_H + (1+\epsilon_H)a_{K,T}\} + \zeta_{K_0}, \tag{8}$$

and consequently, writing $\boldsymbol{\beta}^\dagger = (\boldsymbol{\beta}_1^{\dagger\top}, \ldots, \boldsymbol{\beta}_K^{\dagger\top})^\top \in \mathbb{R}^q$*, for every trajectory and visit* t*,*

$$\left|\mu_{it} - Z_{it}^\top \boldsymbol{\beta}^\dagger\right| \le t\Delta_{K,H,T} \le T\Delta_{K,H,T}. \tag{9}$$

The bound composes *three assumed error levels (feature completeness* ϵ_H*, time-basis error* $a_{K,T}$*, low-rank residual* ζ_{K_0}*) and propagates them through cumulative summation. It derives no spline or ReLU approximation rate; its content is the separation of the three sources and the explicit* $t \le T$ *amplification.*

Theorem A3 (Exact risk decomposition and structure-free bias; Theorem 1 restated)**.** *Under* ***(A1)*** *alone, conditionally on* $\mathcal{F}_{A,X_B}$ *and for* any *conditional mean* μ_B*,*

$$\mathcal{R}_{\text{in}}(\hat{\mu}) = b_K^2 + \frac{\operatorname{tr}(P_K\Sigma_B)}{N_B} \le b_K^2 + \frac{\nu_T r_K}{n_B T}, \qquad b_K := \frac{\|(I-P_K)\mu_B\|_2}{\sqrt{N_B}}, \tag{10}$$

where $P_K = \mathbf{Z}_B\mathbf{Z}_B^+$*,* $\Sigma_B = \operatorname{blockdiag}(\Sigma_i : i \in \mathcal{I}_B)$ *and* $r_K = \operatorname{rank}(\mathbf{Z}_B) \le K(H+1)$*. No structural assumption and no lower eigenvalue condition on* $\mathbf{Z}_B^\top\mathbf{Z}_B$ *enters this identity.*

Corollary A1 (Structural control of the bias)**.** *If* ***(A2)–(A4)*** *also hold, then on* $\mathcal{E}_{A,H}$*,* $b_K \le T\Delta_{K,H,T}$*, hence*

$$\mathcal{R}_{\text{in}}(\hat{\mu}) \le T^2\Delta_{K,H,T}^2 + \frac{\nu_T r_K}{n_B T} \le T^2\Delta_{K,H,T}^2 + \frac{\nu_T K(H+1)}{n_B T}. \tag{11}$$

Thus ***(A2)–(A4)*** *are* sufficient, not necessary*: they only bound the always-defined* b_K*, which is itself estimable from fold-*B *residuals up to a noise-trace correction, since* $\mathbb{E}[\|(I-P_K)Y_B\|_2^2 \mid \mathcal{F}_{A,X_B}] = \|(I-P_K)\mu_B\|_2^2 + \operatorname{tr}((I-P_K)\Sigma_B)$*.*

Corollary A2 (Bound-optimal smooth-basis size)**.** *Suppose a finite smooth-basis candidate set (*$3 \le K < T$*) satisfies* $a_{K,T} \le C_B L K^{-\bar{\alpha}}$ *for constants* $C_B > 0$*,* $\bar{\alpha} > 0$*, and* $\epsilon_H \le 1$*. Holding* H, T *fixed, the* K*-dependent part of* (11) *is at most*

$$U(K) = D_{T,H}K^{-2\bar{\alpha}} + V_{n,T,H}K, \quad D_{T,H} = 3T^2K_0^2(1+\epsilon_H)^2C_B^2L^2, \quad V_{n,T,H} = \frac{\nu_T(H+1)}{n_B T}, \tag{12}$$

plus the K*-independent floor* $3T^2(K_0^2L^2\epsilon_H^2 + \zeta_{K_0}^2)$*. If* $D_{T,H} > 0$ *and* $V_{n,T,H} > 0$ *the positive-real minimizer is*

$$K_{\text{cont}} = \Big(\frac{2\bar{\alpha}D_{T,H}}{V_{n,T,H}}\Big)^{1/(2\bar{\alpha}+1)} = \Big(\frac{6\bar{\alpha}K_0^2(1+\epsilon_H)^2C_B^2L^2\, n_B T^3}{\nu_T(H+1)}\Big)^{1/(2\bar{\alpha}+1)}, \tag{13}$$

and the bound-optimal candidate is an admissible value adjacent to K_{cont} *on either side, after endpoint clipping. The continuous minimizer is interior,* $K_{\text{cont}} < T$*, if and only if*

$$\frac{n_B}{\nu_T(H+1)} < \frac{T^{2\bar{\alpha}-2}}{6\bar{\alpha}K_0^2(1+\epsilon_H)^2C_B^2L^2}. \tag{14}$$

For the finite candidate set this is necessary but not sufficient *for a smooth candidate to beat the indicator endpoint; that discrete comparison holds iff* $\min_{3\le K\le T-1} U(K) < U(T)$*, for which a sufficient condition is that some admissible* K *lies in* (K_{cont}, T)*. If* $D_{T,H} = 0 < V_{n,T,H}$ *the smallest candidate is optimal; if* $V_{n,T,H} = 0 < D_{T,H}$ *the largest; if both vanish every candidate ties. These are statements about the upper bound* U*, not assertions that the same* K *minimizes actual risk.*

Corollary A3 (What the coefficient count does and does not imply)**.** *For a frozen* K*-basis design satisfying* ***(A1)****,*

$$\frac{\operatorname{tr}(P_K\Sigma_B)}{N_B} \le \frac{\nu_T r_K}{N_B} \le \frac{\nu_T K(H+1)}{n_B T},$$

so the dimension-based upper bound *at* K *is* K/T *times the corresponding bound at the indicator endpoint* $K = T$*. This does not imply that the two actual traces have ratio* K/T*, nor that smoothing reduces total risk. A sufficient bound-level condition for a smooth candidate* $K < T$ *to have a smaller upper bound than the indicator candidate is*

$$T^2\Delta_{K,H,T}^2 + \frac{\nu_T K(H+1)}{n_B T} < T^2\Delta_{T,H,T}^2 + \frac{\nu_T T(H+1)}{n_B T}, \tag{15}$$

where $a_{T,T} = 0$ *gives* $\Delta_{T,H,T} = K_0 L\epsilon_H + \zeta_{K_0}$*. No universal comparison with the sub-network model follows, because it learns* T *different bodies and therefore has a different approximation error and a different projection.*

Reconciling the T^2 factor with "advantage grows with T". The bias carries a worst-case T^2 from cumulative summation, so it *increases* in T, while the dimension-variance term $\nu_T K(H+1)/(n_B T) = \sigma_\varepsilon^2 \bar{\kappa}_T K(H+1)/(n_B T)$ decreases in T only when $\bar{\kappa}_T = o(T)$; strong within-subject dependence ($\bar{\kappa}_T \asymp T$) cancels the $1/T$. By (14) the interior-optimum region widens with T exactly when $\bar{\alpha} > 1$ and shrinks, eventually forcing $K_{\text{cont}} \geq T$ as n_B grows, when $\bar{\alpha} \leq 1$. The theory therefore supports "advantage grows with T" only in the smooth-coefficient, weak-dependence regime. Our simulations fix T=5, so they neither confirm nor refute this boundary; the T-sweep in Appendix C.8 tests it directly.

A.4 Proofs

Proof of Theorem A1. Under the additive target, $\mu_{it} = \sum_{s \leq t} g_s(\mathbf{X}_{is})$ and $\mu_{i,t-1} = \sum_{s \leq t-1} g_s(\mathbf{X}_{is})$ for $t \geq 2$, where $\mu_{i,t-1}$ is the conditional mean given $(\mathbf{X}_{i1}, \ldots, \mathbf{X}_{i,t-1})$. As functions of the covariate history their difference is

$$\mu_{it} - \mu_{i,t-1} = \sum_{s \leq t} g_s(\mathbf{X}_{is}) - \sum_{s \leq t-1} g_s(\mathbf{X}_{is}) = g_t(\mathbf{X}_{it}),$$

which depends on $\mathbf{X}_{it}$ only; integrability of g_t makes each conditional mean well defined. For $t = 1$, $\mu_{i1} = g_1(\mathbf{X}_{i1})$. The displayed equalities hold $P_{\mathbf{X}_{it}}$-almost surely, so g_t is determined $P_{\mathbf{X}_{it}}$-almost everywhere; two versions differing only on a $P_{\mathbf{X}_{it}}$-null set produce identical conditional means, and nondegenerate support does not by itself distinguish them, so identification is up to almost-everywhere equivalence. If in addition each g_s is continuous, then two continuous functions agreeing $P_{\mathbf{X}_{it}}$-a.e. agree on $\operatorname{supp}(P_{\mathbf{X}_{it}})$ (they coincide on a dense subset of the support and are continuous), which upgrades the identification to pointwise on that support. Finally, a shift $g_s \mapsto g_s + c_s$ by constants changes $\mu_{i1} = g_1 + c_1$ unless $c_1 = 0$; then $\mu_{i2} - \mu_{i1} = g_2 + c_2$ forces $c_2 = 0$, and inductively $c_s = 0$ for all s. Hence no centering constraint is needed for uniqueness. □

Proof of Theorem A2. On $\mathcal{E}_{A,H}$ choose γ_r as in (6), write $v_r(\mathbf{x}) = \tilde{\tilde{\boldsymbol{\phi}}}(\mathbf{x})^\top \gamma_r$, and choose $\theta_r^{K,T}$ as in (5). For each k define $\boldsymbol{\beta}_k^\dagger = \sum_{r=1}^{K_0} \theta_{rk}^{K,T} \gamma_r$. For any visit s and covariate value $\mathbf{x}$, linearity gives

$$\sum_{k=1}^{K} B_k^{K,T}(s) \tilde{\tilde{\boldsymbol{\phi}}}(\mathbf{x})^\top \boldsymbol{\beta}_k^\dagger = \sum_{r=1}^{K_0} \{B^{K,T}(s)^\top \theta_r^{K,T}\} v_r(\mathbf{x}).$$

By **(A2)**, $g_s(\mathbf{x}) = \sum_r c_r(u_s) h_r(\mathbf{x}) + \delta_s(\mathbf{x})$. Adding and subtracting $\sum_r c_r(u_s) v_r(\mathbf{x})$, the triangle inequality yields

$$\begin{aligned}
&\Big| g_s(\mathbf{x}) - \sum_{k=1}^{K} B_k^{K,T}(s) \tilde{\tilde{\boldsymbol{\phi}}}(\mathbf{x})^\top \boldsymbol{\beta}_k^\dagger \Big| \\
&\quad \leq \sum_{r=1}^{K_0} |c_r(u_s)|\, |h_r(\mathbf{x}) - v_r(\mathbf{x})| + \sum_{r=1}^{K_0} \big| c_r(u_s) - B^{K,T}(s)^\top \theta_r^{K,T} \big|\, |v_r(\mathbf{x})| + |\delta_s(\mathbf{x})|.
\end{aligned}$$

Assumption **(A2)** bounds the first coefficient by L and gives $|\delta_s(\mathbf{x})| \leq \zeta_{K_0}$; **(A4)** gives $|h_r - v_r| \leq \epsilon_H$ and, since $\|h_r\|_\infty \leq 1$, also $|v_r| \leq 1 + \epsilon_H$; **(A3)** bounds the remaining coefficient error by $a_{K,T}$. Summing the K_0 terms and adding ζ_{K_0} proves (8). Finally

$$\mu_{it} - Z_{it}^\top \boldsymbol{\beta}^\dagger = \sum_{s \leq t} \Big\{ g_s(\mathbf{X}_{is}) - \sum_{k=1}^{K} B_k^{K,T}(s) \tilde{\tilde{\boldsymbol{\phi}}}(\mathbf{X}_{is})^\top \boldsymbol{\beta}_k^\dagger \Big\};$$

applying (8) to every summand and using the triangle inequality proves (9). □

Proof of Theorem A3. Condition on $\mathcal{F}_{A,X_B}$; the matrices $\mathbf{Z}_B$, P_K, Σ_B and the vector μ_B are then fixed. The Moore–Penrose identities imply that $P_K = \mathbf{Z}_B \mathbf{Z}_B^+$ is the symmetric orthogonal projection onto $\operatorname{col}(\mathbf{Z}_B)$. Since $Y_B = \mu_B + \varepsilon_B$,

$$\hat{\mu}_B - \mu_B = P_K Y_B - \mu_B = -(I - P_K)\mu_B + P_K \varepsilon_B.$$

By independence across subjects and **(A1)**, each fold-B subject is independent of fold A and of the other subjects, so $\mathbb{E}(\varepsilon_B \mid \mathcal{F}_{A,X_B}) = 0$ and $\mathrm{Cov}(\varepsilon_B \mid \mathcal{F}_{A,X_B}) = \Sigma_B$; the cross term therefore has zero conditional expectation. Moreover

$$\mathbb{E}(\|P_K\varepsilon_B\|_2^2 \mid \mathcal{F}_{A,X_B}) = \mathrm{tr}(P_K^\top P_K \Sigma_B) = \mathrm{tr}(P_K\Sigma_B),$$

because $P_K^\top P_K = P_K$. Dividing by N_B gives the identity in (10) with $b_K^2 = \|(I-P_K)\mu_B\|_2^2/N_B$; nothing beyond **(A1)** was used and μ_B was arbitrary. For the variance term, $0 \preceq P_K \preceq I$ and $\Sigma_B \preceq \nu_T I$, so

$$\mathrm{tr}(P_K\Sigma_B) = \mathrm{tr}(\Sigma_B^{1/2} P_K \Sigma_B^{1/2}) \le \nu_T \,\mathrm{tr}(P_K) = \nu_T r_K \le \nu_T K(H+1),$$

and $N_B = n_B T$, which yields the displayed inequality. □

Proof of Corollary A1. On $\mathcal{E}_{A,H}$ let $\boldsymbol{\beta}^\dagger$ be supplied by Theorem A2. Orthogonal projection is the best Euclidean approximation in $\mathrm{col}(\mathbf{Z}_B)$, hence

$$\sqrt{N_B}\, b_K = \|(I-P_K)\mu_B\|_2 \le \|\mu_B - \mathbf{Z}_B\boldsymbol{\beta}^\dagger\|_2 \le \sqrt{N_B}\, T\Delta_{K,H,T},$$

the last step entrywise from (9). Substituting this and $\mathrm{tr}(P_K\Sigma_B) \le \nu_T r_K \le \nu_T K(H+1)$ into (10) proves both inequalities in (11). The estimability statement is the tower property applied to $\|(I-P_K)Y_B\|_2^2 = \|(I-P_K)(\mu_B+\varepsilon_B)\|_2^2$, whose conditional expectation is $\|(I-P_K)\mu_B\|_2^2 + \mathrm{tr}((I-P_K)\Sigma_B)$ because $I-P_K$ is an $\mathcal{F}_{A,X_B}$-measurable projection and the cross term vanishes as above. □

Proof of Corollary A2. By (8), $\Delta_{K,H,T} = K_0 L\epsilon_H + K_0(1+\epsilon_H)a_{K,T} + \zeta_{K_0}$. The inequality $(x+y+z)^2 \le 3(x^2+y^2+z^2)$ together with $a_{K,T} \le C_B L K^{-\bar\alpha}$ gives

$$T^2\Delta_{K,H,T}^2 \le 3T^2\big(K_0^2L^2\epsilon_H^2 + \zeta_{K_0}^2\big) + 3T^2K_0^2(1+\epsilon_H)^2C_B^2L^2K^{-2\bar\alpha}.$$

Adding the dimension variance bound in (11) gives the stated K-independent floor plus $U(K)$. Suppose first $D_{T,H} > 0$ and $V_{n,T,H} > 0$. For real $K > 0$,

$$U'(K) = -2\bar\alpha D_{T,H}K^{-2\bar\alpha-1} + V_{n,T,H}, \qquad U''(K) = 2\bar\alpha(2\bar\alpha+1)D_{T,H}K^{-2\bar\alpha-2} > 0,$$

so U is strictly convex and its unique stationary point is its global minimizer. Solving $U'(K) = 0$ gives the first equality in (13); substituting $D_{T,H}, V_{n,T,H}$ gives the second. On a finite candidate set, strict decrease before and strict increase after K_{cont} show that only the immediately adjacent admissible candidates on its two sides, after endpoint clipping, need be compared. The continuous minimizer is interior iff $2\bar\alpha D_{T,H}/V_{n,T,H} < T^{2\bar\alpha+1}$, which rearranges to (14); this is a statement about the real minimizer of U, *not* about the discrete comparison with $U(T)$. *Case $K_{\mathrm{cont}} < T$:* the endpoint T lies on the increasing branch, so by strict convexity $\{K > 0 : U(K) < U(T)\} = (K_-, T)$ where $K_- < K_{\mathrm{cont}}$ is the unique other solution of $U(K_-) = U(T)$; thus a smooth candidate beats $U(T)$ iff $\min_{3\le K\le T-1} U(K) < U(T)$, and an admissible $K \in (K_{\mathrm{cont}}, T)$ is sufficient because U is increasing there. It remains possible that every admissible smooth $K \le K_{\mathrm{cont}}$ has $U(K) > U(T)$, so $K_{\mathrm{cont}} < T$ alone does not imply a smooth candidate wins. *Case $K_{\mathrm{cont}} \ge T$: U* is nonincreasing on $(0, T]$, so $U(T) \le U(K)$ for every admissible $K < T$. If $D_{T,H} = 0 < V_{n,T,H}$ then $U(K) = V_{n,T,H}K$ is strictly increasing and the smallest candidate is optimal; if $V_{n,T,H} = 0 < D_{T,H}$ then $U(K) = D_{T,H}K^{-2\bar\alpha}$ is strictly decreasing and the largest is optimal; if both coefficients vanish, $U(K) = 0$ for every candidate. □

Proof of Corollary A3. The two variance inequalities are exactly the trace and rank inequalities established in the proof of Theorem A3. Writing $\mathcal{V}_K = \nu_T K(H+1)/(n_B T)$ for the dimension-based upper bound gives $\mathcal{V}_K = (K/T)\mathcal{V}_T$, including when $\nu_T = 0$. This identity applies only to the dimension-based upper bounds; the actual terms are $\mathrm{tr}(P_K\Sigma_B)/N_B$ and $\mathrm{tr}(P_T\Sigma_B)/N_B$, and because the projections may have different ranks and orientations relative to Σ_B, the preceding inequalities imply no ratio between these traces. Applying (11) separately at K and T shows that (15) is sufficient for the displayed upper bound at K to be smaller. For the indicator basis, choose $\theta_{rk}^{T,T} = c_r(u_k)$, which makes $B^{T,T}(s)^\top\theta_r^{T,T} = c_r(u_s)$ and hence $a_{T,T} = 0$; substitution in (8) gives $\Delta_{T,H,T} = K_0 L\epsilon_H + \zeta_{K_0}$. □

Scope of the results. The exact identity (10) holds for any conditional mean and separates approximation to the realized column space, measured by b_K, from noise projected into that space; **(A2)–(A4)** enter only to bound b_K, and b_K is itself estimable, so the structural premises are checkable in effect even though the population factorisation is not. A nonzero ζ_{K_0} propagates as an irreducible bias. Assumption **(A4)** bundles representation and optimization success, so the size of ϵ_H attained by stochastic-gradient training remains open. The estimator analyzed is a single sample split, whereas the deployed SMOOTHCUM trains body and heads jointly on all subjects, and OLS-refit agreement does not prove their equivalence. Finally, low time rank remains load-bearing: high-rank time-varying shapes and genuine cross-time interactions inflate ζ_{K_0} and b_K and are not covered, which is precisely why the combined pool of Appendix D carries a sequence model.

A.5 THEORY FOR THE SHARED BODY

Theorem 1 concerns the final layer only; the shared-body effect we measure in §6.2 is a statement about representation learning, and its theory comes from the multitask literature. Two observations fix its scope.

The body effect is intrinsically a nonlinear-body phenomenon. If the body is linear, $\boldsymbol{\phi}(\mathbf{x}) = Wx$ with $W \in \mathbb{R}^{H\times d}$ shared, the multiplicative head collapses it into the coefficients: $g(s,\mathbf{x}) = \sum_k B_k(s)(\boldsymbol{w}_k^\top W x + b_k) = \sum_k B_k(s)(\boldsymbol{v}_k^\top x + b_k)$ with $\boldsymbol{v}_k = W^\top \boldsymbol{w}_k$, so for $H \geq K$ the shared-linear-body class equals the class with $K(d{+}1)$ free coefficients and Theorem 1 applies verbatim with $q = K(d{+}1)$; at $K{=}T$ the shared-linear-body and per-visit-linear-body classes coincide. Any variance gap at matched coefficient count must therefore come from nonlinearity of the body—consistent with the measurements, which use ReLU bodies.

For nonlinear bodies the sharing benefit is the multitask representation effect of Maurer et al. (2016), imported with the following mapping rather than claimed as a new result.

Proposition A1 (shared-body rate; adapted from Maurer et al., 2016)**.** *Identify the T per-visit regressions of Theorem A1 with T tasks sharing the body class Φ and drawing their samples from the same n_B subjects; let the per-task heads range over a class $\mathcal{F}$ with the loss bounded and Lipschitz. Then, with independence across subjects and within-subject dependence absorbed into constants via the envelope of* (A1)*, the excess average risk of empirical risk minimization with a shared body is bounded by*

$$c_1 \frac{C(\Phi)}{\sqrt{n_B T}} + c_2 \frac{C(\mathcal{F})}{\sqrt{n_B}} + O\Big(\sqrt{\tfrac{\log(1/\delta)}{n_B}}\Big),$$

where $C(\cdot)$ are the Gaussian-average complexities of Maurer et al. (2016), whereas with T per-visit bodies the first term is $c_1\, C(\Phi)/\sqrt{n_B}$: sharing divides the representation term by $\sqrt{T}$.

Three qualifications apply. The samples of the T tasks here are the same subjects, not independent task datasets; independence across subjects supports the concentration step, but a dependence-sharp version of the constants is open. The bound describes empirical risk minimization, not the optimization path of joint training. And in our architecture the heads are additionally tied by the basis, so the second term is itself controlled by Theorem 1's q-linear form rather than by $C(\mathcal{F})$ alone; the two results overlap rather than compose exactly. The proposition's role matches the head identity's: it names the source of the variance reduction and its scaling ($\sqrt{T}$ for the body, q for the head), both of which §6.2 then measures.

Table 3: Estimators of the cumulative mean (1) compared in this paper. "Cumulative form" records whether the estimator imposes the sum over past visits in (1); "restriction on g_s" is the functional form assumed within a visit. The last column is the integrated squared error (ISE) averaged over the 150 settings of the simulation grid of §6.3 (Table 2); the two recent sequence architectures are evaluated in the scale study of Appendix C.4. Contemporaneous estimators, which use $\mathbf{X}_{it}$ only, are far behind all of these (ISE 46–50) and are omitted.

Estimator	Reference	Cumulative form	Restriction on g_s	ISE
DLNM	Gasparrini et al. (2010)	yes, by lag $t-s$	additive across covariates	11.16
Functional linear model	Malfait & Ramsay (2003)	yes	linear in $\mathbf{X}_{is}$	33.03
Two-step smoothing	Fan & Zhang (2000)	yes	none; T separate networks, smoothed across visits after fitting	25.08
Sub-network estimator	direct encoding of (1)	yes	none; T separate networks	23.86
GRU	Cho et al. (2014)	no	none	32.42
Causal Transformer	Vaswani et al. (2017)	no	none	33.16
TCN	Bai et al. (2018)	no	none	25.72
Selective state-space model	Gu & Dao (2024)	no	none	Appendix C.4
Patch transformer	Nie et al. (2023)	no	none	Appendix C.4
Gradient boosting (GBM)	Friedman (2001)	no	none; history as concatenated features	19.92
SMOOTHCUM	this paper	yes, by visit s	none; one shared network, rank-K spline output layer	**15.99**

B ESTIMATORS COMPARED

We list in Table 3 every estimator compared in the paper, the structure each imposes on (1), its reference, and its error averaged over the comparison grid of §6.3.

C SIMULATION DETAILS AND FULL RANKINGS

Residual structures on the comparison grid are `independent`, `exchangeable`, `ar1`, `rand_intercept` and `nonstationary`, all at $\rho_\varepsilon = 0.5$, with σ_ε scaled to $\mathrm{SNR} = 2$ so settings are comparable; two stress structures (exchangeable $\rho_\varepsilon = 0.9$; exchangeable $\rho_\varepsilon = 0.8$ with time-varying noise SD) form a separate 60-setting slice on which the method ranking is unchanged. All neural methods tune width $\times$ learning rate by subject-level validation inside the training set; GRU, Transformer, TCN, SSM and PatchTST additionally use validation-based early stopping, so a deficit reflects inductive bias rather than under- or over-training.

Table 4: Comparison grid split by individual signal pattern (test ISE, mean over settings); Table 1 aggregates the four g_{base} columns as the additive class. `interaction` uses the non-additive g_{int} ($\pi_{\mathrm{add}} = 0.27$); the rest use g_{base} ($\pi_{\mathrm{add}} = 0.98$) under four time-varying coefficients.

Method	const	altern	decay	increas	inter
DLNM	**11.37**	**12.59**	**8.87**	**6.87**	16.13
SMOOTHCUM	25.67	23.42	15.46	6.94	**8.44**
GBM	31.88	25.16	17.98	9.82	14.76
Subnet-Cum	39.56	31.33	24.01	11.06	13.32
GRU	56.02	44.80	31.35	15.89	14.04

The decomposition study (§6.2) draws one evaluation set of 1,000 subjects per setting with a fixed seed, refits every estimator on 60 fresh training draws, and computes $\mathrm{bias}^2(\mathbf{x}) = (\bar{\hat{\mu}}(\mathbf{x}) - \mu(\mathbf{x}))^2$ and $\mathrm{var}(\mathbf{x}) = \widehat{\mathrm{Var}}_r\, \hat{\mu}_r(\mathbf{x})$ pointwise, with variance divisor $R - 1 = 59$. With divisor R the sum $\overline{\mathrm{bias}^2} + \overline{\mathrm{var}}$ equals the empirical ISE exactly; with divisor $R - 1$ the variance term is larger by the factor $60/59$, below the rounding of the reported figures. Hyperparameters are fixed at the operating point the tuned pipeline selects (width 16, two hidden layers, learning rate 3×10^{-3}, 400 epochs), so $q = K(H+1)$ is exact; at an untuned corner (width 8, learning rate 10^{-3}) both estimators are bias-dominated and the variance reduction is invisible, which is why the operating point is stated.

Table 5: Decomposition study. *Left:* reference setting (T=5, n=500, SNR 2): bias^2 and variance across K, with the sub-network estimator at the far right. *Right:* fitted slope of variance on $q = K(H+1)$ within SMOOTHCUM and the variance ratio at equal q (sub-network variance over SMOOTHCUM variance at K=T, the column headed "body") across settings.

	K=1	2	3	4	5	subnet
bias^2	6.78	7.20	7.27	7.86	8.15	10.13
variance	4.67	5.04	5.23	5.78	6.36	14.58
ISE	11.45	12.23	12.50	13.64	14.51	24.71

Setting	slope	body
$T5, n200$	0.063	2.06
$T5, n500$	0.024	2.29
$T5, n2000$	0.014	1.74
$T12, n200$	0.049	2.57
$T12, n500$	0.013	3.14
$T12, n2000$	0.006	1.89
SNR 0.5	0.093	2.56
SNR 8	0.014	2.02

Table 6: Smoothness sweep (T=12, n=500, 40 replicates): $a_s = 1 + 0.8\sin(2\pi\nu s/T)$. K^* is selected by cross-validation; the ratio is sub-network ISE over SMOOTHCUM ISE at K^*.

ν	K^*	SMOOTHCUM ISE	sub-network ISE	ratio
0.5	3	93.7	148.7	1.59
1	8	64.0	103.7	1.62
2	12	57.7	90.0	1.56
4	12	56.2	84.2	1.50

Table 7: Scale grid: test ISE, 40 replicates per setting. Bold marks the best entry per row; read it with the model-class rule of Appendix C.4 and §6.3: DLNM leads the additive rows, the proposed family (SMOOTHCUM and its input-coordinate variant) the non-additive ones. Reference setting T=5, d=7, n=500; one factor moves per row, plus a joint corner. Bold: best per row.

signal	T	d	n	DLNM	GRU	Trans.	TCN	SSM	Patch	subnet	concat	SMOOTHCUM
const	5	7	500	**7.8**	43.7	44.4	32.2	69.0	41.0	32.2	14.8	17.8
const	5	7	2000	**4.4**	29.3	19.5	22.8	19.0	20.1	16.3	9.3	11.3
const	5	20	500	**10.4**	49.0	48.5	33.6	86.3	52.5	39.0	28.7	29.8
const	5	50	500	**23.7**	62.5	70.9	48.8	105.9	69.7	60.3	43.9	43.8
const	12	7	500	**16.7**	134.3	110.1	159.6	105.4	114.3	80.7	37.5	47.4
const	12	20	2000	**10.2**	135.2	69.8	155.0	105.8	78.5	83.7	46.8	48.4
const	24	7	500	**21.6**	435.4	188.3	537.2	170.7	244.3	178.6	91.3	106.2
inter	5	7	500	12.4	10.5	9.7	10.5	13.5	11.3	9.7	**5.4**	5.8
inter	5	7	2000	12.7	7.5	5.0	7.0	5.3	7.2	5.3	**2.6**	2.8
inter	5	20	500	14.7	13.7	14.1	13.3	17.2	14.2	16.5	**11.0**	11.5
inter	5	50	500	22.0	17.0	17.2	**17.0**	18.1	17.1	23.3	17.2	17.5
inter	12	7	500	29.6	25.7	22.0	30.9	25.6	28.1	26.8	**9.2**	9.6
inter	12	20	2000	28.6	25.6	17.8	30.9	28.8	25.1	28.7	**9.3**	9.9
inter	24	7	500	57.9	50.9	42.5	72.9	34.1	56.9	58.5	**18.1**	18.9

The two current baselines are implemented as follows, both causal so that $\hat{\mu}_{it}$ uses $\mathbf{X}_{i1}, \ldots, \mathbf{X}_{it}$ only. The selective SSM uses a learned negative diagonal A (log-parameterised, S4D-style spread

of decay rates), input-dependent $\Delta_t = \mathrm{softplus}(W_\Delta x_t)$, B_t, C_t, a silu-gated output branch, hidden sizes $\{16, 32\}$, one block, state dimension 8, exact sequential recurrence. The patch transformer uses per-channel convolutional patch embedding (patch length 3, embedding dimension 4 per channel, stride 1, left-padded), channel embeddings concatenated and projected to $d_{\mathrm{model}} = 32$, two encoder layers, four heads, causal mask, sinusoidal positions. Both train with the shared budget (400 epochs, learning rates $\{10^{-3}, 3 \times 10^{-3}\}$) and early stopping.

The hierarchical pairwise-merge baseline of Appendix C.4 embeds each visit with a shared body, then folds the prefix $e_1, \ldots, e_t$ pairwise up a binary tree with one shared merge network (odd node carried), predicting from the root, so it is causal; widths $\{16, 32\}$, same budget and early stopping. On the reference settings (40 replicates, matched to Table 7) its ISE against SMOOTHCUM's is 23.4/17.8 and 45.5/47.4 on the additive signal at T=5/12, and 7.4/5.8 and 14.2/9.6 on the non-additive signal: competitive with the best sequence models, behind the structured estimator throughout.

The nuisance-dimension analysis of §6.3 uses three additional protocols. The degradation factors compare each method's ISE at d=50 to d=7 at T=5, n=500. The width check reruns the $d \in \{20, 50\}$ settings with the cumulative estimators' width grid extended to $\{8, 16, 32, 64\}$, matching the sequence models' ceiling. The per-visit analysis records ISE separately at each visit index and reports $R_t = \mathrm{ISE}_t(d{=}50)/\mathrm{ISE}_t(d{=}7)$; for the neural sum-over-visits estimators the mean slope of R_t in t is -0.01 (penalty flat in t), while for DLNM it is $+0.07$ to $+0.15$ per visit.

C.1 DATA-GENERATING MECHANISM

Covariates are $\mathbf{X}_{it} \sim N(0, \Sigma_X)$, $(\Sigma_X)_{jk} = 0.5^{|j-k|}$, with temporal dependence $\mathbf{X}_{it} = \rho_X \mathbf{X}_{i,t-1} + \sqrt{1-\rho_X^2} Z_{it}$, $Z_{it} \sim N(0, \Sigma_X)$; $d = 7$ (all active), $T = 5$. The cumulative mean is $\mu_{it} = \sum_{s \le t} a_s g(\mathbf{X}_{is})$ with

$$g_{\mathrm{base}}(\mathbf{x}) = x_1^3 + x_2^2 + \sin x_3 + |x_4| + x_5 x_6 + 0.5 e^{-x_7^2}, \qquad g_{\mathrm{int}}(\mathbf{x}) = x_1 x_5^2 + \sin(x_2 x_7) + x_3 x_6 |x_4|.$$

Time weights a_s give the five signal patterns: `constant` ($a_s = 1$), `decaying` (0.8^{s-1}), `increasing` (s/T), `alternating` ($\{1, 0.5, 1.2, 0.7\}$ cycled) (all with g_{base}) and `interaction` ($a_s = 1$ with g_{int}). The residual realises one of five within-subject structures (`independent`, `exchangeable`, `ar1`, `rand_intercept`, `nonstationary`; $\rho_\varepsilon = 0.5$), with σ_ε auto-scaled to SNR $= \mathrm{Var}(\mu_{it})/\sigma_\varepsilon^2 = 2$, where σ_ε^2 is the residual variance, so settings are comparable. Two *stress* structures, `exch_high` (exchangeable $\rho_\varepsilon = 0.9$) and `hetero_exch` (exchangeable $\rho_\varepsilon = 0.8$ with noise SD ranging 0.5–2.5$\times$ across time), are run as a separate slice. The main grid crosses 5 signals $\times$ 5 residual structures $\times\, n \in \{200, 500\} \times \rho_X \in \{0, 0.3, 0.6\}$ (150 settings), 200 Monte-Carlo replicates each; the stress slice adds 60 settings. All 30,000 main-grid replicates ran to completion with *zero* method failures.

Why the interaction signal uses non-adjacent products. Because $\Sigma_X = 0.5^{|j-k|}$ makes adjacent covariates correlated, adjacent products retain large additive projections that a spline absorbs. The products in g_{int} pair covariates at lag ≥ 3 (cross-correlation ≤ 0.125), where the best additive fit captures an additive share of only 0.27. Covariate feature learning is therefore genuinely required, and the DLNM/SMOOTHCUM contrast is not an artifact of a weak additive baseline.

The additive projections, measured. Neither per-visit contribution is described by assertion: regressing the noiseless signal on a rich additive cubic-spline basis (12 knots per covariate) over 5×10^5 draws of $x \sim N(0, \Sigma_X)$ gives additive share $\pi_{\mathrm{add}} = 0.98$ for g_{base} and 0.27 for g_{int} (`scripts_cum/gbase_additive_r2.py`). g_{base} is thus *predominantly* (not exactly) additive: its one interaction $x_5 x_6$ pairs adjacent covariates whose correlation (0.5) leaves most of the term inside the additive span. This is deliberate: an exactly additive signal would make the DLNM's covariate restriction correct by construction, whereas the question the grid asks is how the families fare when the restriction is approximately right.

C.2 THE K-SWEEP AND THE TWO-WAY DECOMPOSITION

The time basis is *fixed*: a quadratic B-spline on $[1, T]$ evaluated once at $s = 1, \ldots, T$ and held constant while the K heads are trained. Its two extremes are members of the same family, so

sweeping K from 1 to T answers "does smoothing help?" *within one model*, holding the shared body, the training budget and the cumulative structure fixed. We report the sweep over the full 150-setting grid for every K in Table 8.

Table 8: Test ISE mean (Monte-Carlo SD over 200 replicates, averaged across settings) versus the number of time-basis functions K, over the *full* 150-setting grid for every K. K=1 is the constant basis (cumulative sum-pooling, no time variation); K=T=5 is the per-time indicator basis (no smoothing). Bold = best.

Method	$K=1$ (constant)	$K=2$	$K=3$	$K=4$	$K=5$ (indicators)
SMOOTHCUM (proposed)	16.02 (9.90)	**15.54 (10.00)**	15.98 (10.16)	16.74 (10.50)	17.60 (10.74)

Two facts stand out. First, smoothing *does* help: the best interior basis (K=2, ISE 15.54) improves on the no-smoothing indicator basis (K=T=5, 17.60) by 12%, the bias–variance gain the basis is meant to buy, since too many heads spend the $q = K(H+1)$ budget on variance. Second, the minimum is *shallow*: $K \in \{1, 2, 3\}$ all sit within 0.5 ISE of one another, so the exact basis size barely matters over that range. We keep K=3 throughout the main grid (within 3% of the K=2 optimum), noting that validation would pick K=2. Strikingly, even K=1, pure cumulative sum-pooling with *no* time variation, is already competitive (16.02), because several signal patterns have a time-constant per-visit contribution and the shared body absorbs the rest.

Isolating the two changes. SMOOTHCUM differs from the sub-network baseline in two ways at once: one shared body instead of T independent ones, and $K \ll T$ smooth heads instead of T per-time heads. The sweep separates them with no new machinery, because K=T=5 is the per-time indicator basis: it shares the *same* basis as the sub-network model and differs only in using one shared body. Hence

- **Shared body** (basis held at per-time indicators): Subnet-Cum 23.87 → SMOOTHCUM at K=5, 17.60, a 26% reduction from tying the T cross-sectional problems together through one body;
- **Time basis** (shared body held fixed): K=5 indicators 17.60 → K=2 smooth 15.54, a further 12% from smoothing over time.

The shared body accounts for roughly three-quarters of the total gain and the smooth basis for the remaining quarter.

Basis degree at small K. For $K <$ degree + 1 an open quadratic knot vector degenerates: every basis function vanishes at s=T, so the model would contribute nothing at the final visit. The basis therefore lowers its degree when K is small (K=2 gives piecewise-linear hat functions), which preserves the partition of unity over the whole range. Every K in Table 8 is run over the same 150 settings, so the entries are comparable across K.

C.3 THE CHOICE OF K: THE SMOOTHNESS SWEEP

The cross-validated K adapts to the smoothness of the true time-varying coefficient (Figure 2a). Making the coefficient a sinusoid, $a_s = 1 + 0.8\sin(2\pi\nu s/T)$ at T=12, puts the required basis size under experimental control: a spline with K basis functions can follow roughly K features of the coefficient function, so the approximation term of the risk decomposition predicts that the selected K^* rises with ν. The selected K^* rises $3, 8, 12, 12$ at $\nu = 0.5, 1, 2, 4$, and the ISE curves in Figure 2a show why: at low ν the curve is minimized in the interior and rises on both sides, while at high ν the bias cost of smoothing dominates and the minimum moves to the endpoint.

When the true coefficient is less smooth than the basis can follow, the advantage of the basis is lost but the advantage of the shared network is not. The margin over the sub-network estimator stays between 1.50 and 1.62 across the sweep: at $\nu \geq 2$, the selected K^*=T gives both estimators the same number of coefficients, and the remaining factor of 1.5 is the effect of the shared network,

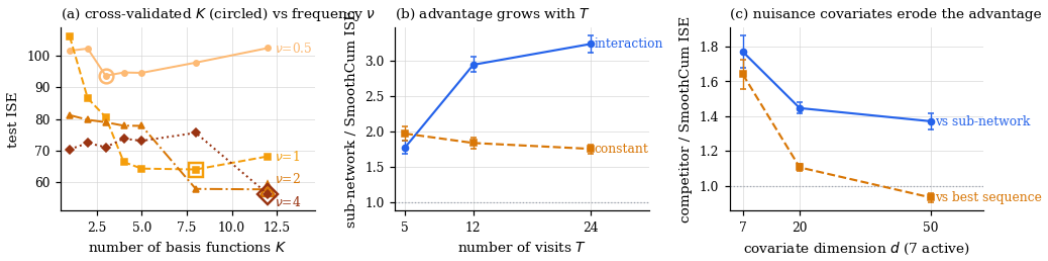

Figure 2: Supporting results (40 replicates per setting). *(a)* Test ISE against the number of basis functions K as the frequency ν of a sinusoidal true time-varying coefficient rises (T=12, n=500); circles mark the cross-validated K^*, which moves from the interior (K^*=3 at ν=0.5) to the per-visit endpoint (K^*=12 at $\nu \geq 2$) as the coefficient becomes less smooth. *(b)* The advantage over the sub-network estimator grows with T on the non-additive covariate function and is flat on the additive one, where the coefficient is exactly representable at K=1. *(c)* On the non-additive covariate function at T=5, adding inactive covariates erodes the advantage over both the sub-network estimator and the best sequence model; the dotted line marks parity.

so the estimator reduces to the per-visit output layers rather than falling behind them. At the opposite endpoint (constant a_s, exactly representable at K=1) cross-validation selects K^*=1. Both endpoints are why K is chosen by cross-validation and never fixed in advance.

C.4 Scale and current baselines

The advantage survives current sequence architectures and grows with sequence length (Figure 2b). To rule out the objection that the sequence baselines are dated, we implement two current architectures, both causal: a selective state-space model (the S6 mechanism, input-dependent Δ_t, B_t, C_t, exact recurrence) and a patch transformer (stride-1, left-padded patch tokens; channel embeddings concatenated, since strict channel independence cannot represent the interaction targets at all). From the reference setting we sweep $T \in \{5, 12, 24\}$, $d \in \{7, 20, 50\}$, $n \in \{500, 2000\}$ and one joint corner, two signals, 40 replicates (14 settings, 9 methods; Table 7, Appendix C).

The T axis behaves as the variance bound predicts. Replacing T visit-specific output layers by K basis coefficients divides the bound by T/K, so the advantage over the sub-network estimator should grow with T; on non-additive targets it does, at 1.77 ± 0.09, 2.94 ± 0.11, 3.24 ± 0.12 (mean paired ratio $\pm$ SE) for $T = 5, 12, 24$. SmoothCum beats the best sequence model in 13 of the 14 settings, each individually significant (Wilcoxon signed-rank test on paired replicates, pairing the two methods' errors on the same replicate, $p < 0.005$); the one exception, the largest nuisance dimension, is a significant loss (0.93 ± 0.03, $p = 0.004$) and is analyzed in §6.3. The ordering within the sequence models is itself informative: the SSM is the strongest sequence model at $T \geq 12$ but the weakest at T=5 on additive targets, so the recent architectures help exactly where the capacity to model long sequences matters and remain behind wherever the cumulative structure matters; the gap is due to the structure imposed, not to the age of the architecture. A hierarchical pairwise-merge encoder ($h(h(e_1, e_2), h(e_3, e_4))$ over shared per-visit embeddings) confirms this from another direction: competitive with the best sequence baselines, it remains a factor of 1.04 to 1.53 behind SmoothCum (Appendix C).

The variant with time as an input (§4) predicts 0–21% better than the product form across these settings while retaining the full margin over the sub-network and sequence models. This supports the interpretation that the operative structure is the shared network together with the low-dimensional smooth encoding of time, and that the choice between the two ways of entering the time basis is secondary; §6.3 shows that even this difference disappears under irregular observation times. We keep the product form as the primary estimator because it is the form the risk decomposition analyzes, and report both throughout.

C.5 NUISANCE COVARIATES

Inactive nuisance covariates erode the advantage, and the erosion arises in the estimation of the per-visit function, not in the accumulation over visits. Adding inactive covariates ($d : 7 \to 50$, the 7 active ones unchanged) degrades estimators that assemble the prediction as a sum over visits by a factor of 2.59 on average (SMOOTHCUM 2.46–3.00) but sequence models by only 1.56. The penalty is present in full at the first visit and does not grow with the visit index, so it attaches to the per-visit function estimate, not to the accumulation; DLNM is the one method whose penalty compounds (the ratio of ISE at d=50 to ISE at d=7, R, rising from 1.1 at the first visit to 2.1 at the last), consistent with its design matrix growing from 127 to 901 columns. Wider networks do not repair the erosion (with the width grid extended to 64, cross-validation still selects widths 9–15 and the ISE changes by under 2%); dimension reduction does: on the cohort task at d=40 (§6.4), projecting to 10 principal components reduces SMOOTHCUM's error by 21% while moving the sequence models by at most 1.9%.

C.6 SAMPLE SIZE, COVARIATE DEPENDENCE, AND DEPENDENCE STRESS

Table 9: Test ISE mean (SD) by sample size (Monte-Carlo SD over replicates, averaged over settings). The smooth model's advantage over the sub-network model holds at both n and is larger at n=500, as the variance-reduction account predicts.

Method	$n = 200$	$n = 500$
SMOOTHCUM (proposed)	20.28 (13.47)	11.70 (6.55)
Subnet-Cum	27.58 (16.11)	20.13 (8.10)
TwoStep (Fan)	29.76 (15.01)	20.40 (7.81)
GRU	36.85 (22.43)	27.99 (12.55)
DLNM	13.00 (9.80)	9.33 (5.28)

Against the contemporaneous `NN_current`, SMOOTHCUM improves ISE by a mean 31.3 with $\Pr(\text{beat}) = 1.00$ and $p < 10^{-3}$ in *every* setting, and the cumulative advantage widens with covariate temporal dependence: mean ISE $(14.3, 15.8, 21.6)$ for SMOOTHCUM versus $(41.0, 47.3, 57.3)$ for the contemporaneous sieve at $\rho_X = (0, 0.3, 0.6)$. We report the ρ_X split without averaging over the residual structures in Table 10; ISE rises monotonically with ρ_X in every one, since stronger temporal covariate dependence spreads the cumulative signal over a longer, more collinear history.

Table 10: Test ISE mean (SD) over 200 replicates by residual-correlation scenario and covariate temporal dependence ρ_X, at the reference configuration (`constant` signal, n=200, K=3). The monotone increase in ρ_X holds across all signals and both sample sizes.

Scenario	Method	$\rho_X = 0$	$\rho_X = 0.3$	$\rho_X = 0.6$
`independent`	SMOOTHCUM	24.59 (16.52)	27.52 (16.14)	37.56 (33.39)
	DLNM	11.06 (10.30)	12.50 (11.66)	15.41 (14.07)
`exchangeable`	SMOOTHCUM	29.55 (16.87)	33.74 (19.97)	44.71 (30.82)
	DLNM	12.56 (10.80)	14.30 (12.08)	18.16 (14.32)
`ar1`	SMOOTHCUM	28.30 (18.22)	32.13 (20.82)	41.24 (30.95)
	DLNM	12.34 (10.77)	14.04 (12.02)	17.92 (14.33)
`rand_intercept`	SMOOTHCUM	23.90 (14.52)	27.80 (18.22)	35.48 (26.30)
	DLNM	11.08 (10.38)	12.38 (11.86)	15.36 (14.63)
`nonstationary`	SMOOTHCUM	26.59 (14.50)	32.39 (21.74)	40.93 (27.85)
	DLNM	12.32 (10.67)	14.08 (12.35)	17.66 (14.91)

Table 11: Robustness slice: test ISE (mean over settings) under strong / heteroscedastic within-subject dependence. The method ranking is unchanged from the main grid. *Note:* these 60 stress settings predate the expanded roster, so the GRU row is from the fixed-epoch trainer (before validation-based early stopping) and Transformer and TCN were not run here; the SMOOTHCUM and DLNM rows are unaffected.

Method	`exch_high` (ρ_ε=0.9)	`hetero_exch`
DLNM	12.12	12.81
SMOOTHCUM (proposed)	19.51	19.60
GRU	35.07	36.23

Because the final layer of every method here uses working independence, SMOOTHCUM's accuracy is essentially flat across correlation regimes (19.5 at ρ_ε=0.9, 19.6 heteroscedastic, against 16.0 at ρ_ε=0.5 on the main grid, the small rise reflecting the harder DGP rather than the dependence per se). The architecture's advantage is a property of the mean structure it encodes, not of the residual correlation, and does not erode as the correlation is strengthened.

C.7 THE SEQUENCE-MODEL CROSSOVER

Throughout the main study the three sequence models trail the cumulative sieve and even the simpler additive baselines, despite each being trained to its own validation optimum. To rule out under-training as the explanation we generate a mean that mixes an additive-over-time part with a non-additive one,

$$\mu_{it} = (1-w)\widetilde{a}_{it} + w\widetilde{c}_{it}, \qquad a_{it} = \sum_{s \le t} g_{\text{base}}(\mathbf{X}_{is}), \quad c_{it} = \sum_{s=2}^{t} r(\mathbf{X}_{is})r(\mathbf{X}_{i,s-1}), \;\; r(\mathbf{x}) = x_1 + x_3,$$

with $\widetilde{a}_{it}, \widetilde{c}_{it}$ the versions of a_{it}, c_{it} standardized to subject variance, μ_{it} then standardized, and $w \in [0,1]$ the interaction weight, so the signal scale is fixed as w varies. The term c_{it} is a sum of *adjacent-time products*: it depends jointly on $\mathbf{X}_{is}$ and $\mathbf{X}_{i,s-1}$ and hence lies outside the additive-over-time class that SMOOTHCUM, the two-step estimator and DLNM all assume.

Table 12: Confirmatory study: test ISE mean (SD) over 8 replicates versus the interaction weight w (n=500; the mean is standardized to subject variance, so ISE is comparable across w). Bold = best in each row.

w	SMOOTHCUM	Subnet	TwoStep	DLNM	GRU	Transf.	TCN
0.0 (additive)	0.262 (0.081)	0.326 (0.058)	0.277 (0.049)	**0.070 (0.025)**	0.284 (0.062)	0.249 (0.068)	0.326 (0.083)
0.5 (mixed)	0.414 (0.102)	0.534 (0.109)	0.524 (0.069)	0.359 (0.052)	0.431 (0.236)	0.339 (0.066)	**0.301 (0.064)**
0.9 (interaction)	0.686 (0.152)	0.832 (0.189)	0.820 (0.129)	0.724 (0.169)	0.302 (0.276)	0.443 (0.145)	**0.276 (0.086)**

At $w = 0$ the structured methods are well specified and competitive, and the three sequence models merely sit among them with no advantage. As w grows, every additive-structured method degrades monotonically because the adjacent-time product is unrepresentable in its function class, while all three sequence models move the opposite way and overtake every additive-structured method once interaction dominates: at $w = 0.9$ TCN (0.28), GRU (0.30) and Transformer (0.44) are 1.6–2.6× more accurate than SMOOTHCUM (0.69), with TCN (not the GRU) the single best, underscoring that the effect is not tied to one architecture. This is the axis on which no convex combination of DLNM and SMOOTHCUM can help, and it is why the deployment pool of §7 carries a sequence model.

C.8 DOES THE ADVANTAGE GROW WITH T? A DIRECT TEST

The paragraph "Reconciling the T^2 factor with 'advantage grows with T' " in Appendix A, referred to below as the growth-in-T prediction, concedes every simulation in this paper fixes $T = 5$, so its conditional claim (SMOOTHCUM's advantage over the per-time sub-network baseline grows with T when the coefficient path is smooth ($\bar{\alpha} > 1$) and dependence is weak ($\bar{\kappa}_T = o(T)$), and need

not otherwise) is untested. We test it directly with a $T \in \{3, 5, 8, 12\}$ sweep, n=500, ρ_X=0.3, 40 replicates, comparing only the four architectures the growth-in-T prediction concerns (DLNM, GRU, Subnet-Cum, SMOOTHCUM). Two arms satisfy both of its conditions, a smooth time-varying coefficient and weak within-subject dependence: `constant` ($a_s \equiv 1$, exactly representable at any spline order, so $\bar{\alpha}$ is effectively unbounded) and `interaction` (same coefficient path, non-additive in covariates), both under AR(1) ρ_ε=0.5 (weak dependence). Two arms break one condition each: `alternating` (a_s cycling $\{1, 0.5, 1.2, 0.7\}$, a non-smooth coefficient path with small $\bar{\alpha}$) under the same weak dependence, and `constant` under `exch_high` (exchangeable ρ_ε=0.9, the $\bar{\kappa}_T \asymp T$ regime in which the growth-in-T prediction says the $1/T$ term is cancelled).

Table 13: SMOOTHCUM's ISE advantage over the sub-network baseline (Subnet-Cum) as a function of T, mean over 40 replicates, n=500, ρ_X=0.3. Ratio = ISE(Subnet-Cum)/ISE(SMOOTHCUM), > 1 favours SMOOTHCUM; every entry is significant at $p < 10^{-10}$ (paired t, within replicate). The two arms satisfying both conditions of the growth-in-T prediction (smooth path, weak dependence) show the largest, most sustained growth; the two arms that each break one condition (non-smooth coefficient path; strong dependence) show growth that saturates or reverses beyond T=8. Bold = largest ratio in the row.

Arm	condition broken	T=3	T=5	T=8	T=12	ratio(T=12)/ratio(T=3)
`constant`, ar1	none	1.35	1.80	**1.95**	1.81	1.34×
`interaction`, ar1	none	1.29	1.76	2.29	**2.66**	2.06×
`alternating`, ar1	smooth path	1.40	1.49	**1.50**	1.49	1.07×
`constant`, exch_high	weak dependence	1.38	1.59	**1.78**	1.71	1.24×

The pattern matches the growth-in-T prediction. Both conditions-satisfied arms grow through the full range swept, `interaction` the more sharply (ratio more than doubles, $1.29 \to 2.66$, with no sign of saturating by T=12); `constant` grows similarly through T=8 then dips slightly. Both broken-condition arms grow far less: the non-smooth coefficient path (`alternating`) is essentially flat from T=5 on ($1.49 \to 1.50 \to 1.49$, a 7% total change against `constant`'s matched 34%), and strong dependence (`constant`/`exch_high`, otherwise identical to the base `constant` arm) sits below the matched weak-dependence arm at every $T \geq 5$ and grows by a smaller total factor (24% against 34%). Four points per arm cannot establish more than the direction (Spearman $\rho = +0.80$ to $+1.00$ across all four arms, the ceiling attainable at n=4), but the direction, and the contrast between the conditions-satisfied and conditions-broken arms, is exactly what the growth-in-T prediction of Appendix A says. We read this as that prediction tested and supported as stated, not merely left unrefuted.

C.9 PER-SETTING RESULTS

The per-setting means and Monte Carlo standard deviations for all 150 settings of the comparison grid, every method and both scores (ISE and test MSE) are provided as CSV files in the code repository.

D THE COMBINATION STUDY IN FULL

D.1 DESIGN

Six scenarios cross the two per-visit contributions g_{base} and g_{int} of the main text's Eq. (3), g_{base} and g_{int}, with three time-weight patterns a_s, `constant` ($a_s \equiv 1$), `decaying` (0.8^{s-1}), `increasing` (s/T), each at $n \in \{200, 500\}$: 12 settings, 100 replicates each, all at $T = 5$, $d = 7$, AR(1) residuals with $\rho_\varepsilon = 0.5$, $\rho_X = 0.3$, SNR $= 2$, and the same 80/20 subject-level split, so every number is comparable to Appendix C. As a **residual-structure sensitivity analysis**, the (g_{base}, `constant`, n=500) scenario is rerun under all five residual structures (`independent`, `exchangeable`, `ar1`, `rand_intercept`, `nonstationary`), 100 replicates each. Candidates are DLNM, SMOOTHCUM, and a GRU, each trained exactly as in Appendix C with the same light uniform tuning. Weights use $V = 4$ subject-level folds; restricted pools are obtained by support-

ing the simplex grid only on the retained coordinates, so a restricted pool is exactly a coordinate constrained to zero.

D.2 Every setting

Table 14 is the complete per-setting result underlying the deployment claims of the main paper's §7. The combined predictor of that section, the weighted average of DLNM, SmoothCum and the GRU with weights chosen by cross-validation over subject-level folds, is called CumBlend in the tables below; "selection" means choosing the single estimator with the smallest cross-validation error. Two readings are worth making explicit. (i) The combined predictor's advantage over selection is *one-sided*: CumBlend beats selection where no single family dominates and matches it where one does, and never loses significantly. (ii) Equal weights lose badly, because the candidates are too unequal in accuracy for the forecast-combination puzzle to apply. The restricted pools (D+S, D+G, S+G) are reported for completeness; the full pool is the estimator the paper recommends.

Table 14: The combination study, every setting, test ISE, mean over 100 replicates. *Sel* is the CV-selected single family on the same out-of-fold predictions as the weights. Pools are subsets of {D=DLNM, S=SmoothCum, G=GRU}; the full pool D+S+G is CumBlend; *Eq* is equal weights on the full pool. $\hat{w}$ are the fitted full-pool simplex weights. Δ and p are the paired difference of CumBlend against selection and its t-test. Bold = best in row.

Scenario	n	D	S	G	Sel	CumBlend	D+S	D+G	S+G	Eq	$\hat{w}_D$	$\hat{w}_S$	$\hat{w}_G$	Δ (p)
g_{b} `constant`	200	13.148	28.184	55.570	13.148	12.892	**12.796**	13.257	27.666	20.584	0.85	0.13	0.02	−0.26 (0.025)
g_{b} `constant`	500	8.307	19.542	46.020	8.307	8.266	**8.253**	8.327	19.562	15.409	0.88	0.12	0.00	−0.04 (0.450)
g_{b} `decaying`	200	10.636	19.075	31.982	10.636	10.307	**10.225**	10.650	18.962	13.896	0.78	0.18	0.03	−0.33 (0.055)
g_{b} `decaying`	500	6.955	11.119	24.878	6.969	6.420	**6.389**	6.934	11.196	9.274	0.75	0.24	0.01	−0.55 (<0.001)
g_{b} `increasing`	200	7.039	8.218	15.882	7.341	5.841	**5.840**	7.084	8.152	7.356	0.55	0.44	0.00	−1.50 (<0.001)
g_{b} `increasing`	500	5.337	4.513	12.211	4.710	**3.605**	**3.605**	5.329	4.518	4.935	0.47	0.53	0.00	−1.11 (<0.001)
g_{i} `constant`	200	16.245	**10.184**	15.233	10.691	10.381	10.350	14.639	10.264	11.555	0.09	0.66	0.25	−0.31 (0.197)
g_{i} `constant`	500	13.208	5.783	11.092	5.783	5.668	5.718	10.915	**5.658**	7.706	0.04	0.81	0.15	−0.12 (0.001)
g_{i} `decaying`	200	9.681	6.616	8.717	6.927	6.354	6.513	8.373	**6.287**	6.806	0.09	0.66	0.25	−0.57 (<0.001)
g_{i} `decaying`	500	8.885	3.923	7.006	3.923	**3.790**	3.843	6.841	3.795	4.940	0.03	0.78	0.18	−0.13 (<0.001)
g_{i} `increasing`	200	4.453	2.831	3.984	2.862	2.724	2.775	3.895	**2.715**	3.057	0.08	0.68	0.24	−0.14 (<0.001)
g_{i} `increasing`	500	4.060	1.778	3.505	1.778	1.757	1.763	3.473	**1.750**	2.409	0.04	0.81	0.15	−0.02 (0.072)
mean		8.996	10.147	19.673	6.923	**6.500**	6.506	8.310	10.044	8.994				

D.3 Residual-structure sensitivity

Table 15 repeats the (g_{base}, `constant`, n=500) setting under all five residual structures. Under every structure CumBlend is favorable to or statistically indistinguishable from selection (the one nominally adverse difference is +0.006 ISE under `exchangeable`, $p = 0.94$) and equal weights are far worse throughout, so the main design's AR(1) choice is not load-bearing.

Table 15: Residual-structure sensitivity: the (g_{base}, `constant`, n=500) scenario under all five residual structures, test ISE, mean over 100 replicates. Δ and p as in Table 14.

Residual structure	DLNM	SmoothCum	GRU	Select	CumBlend	Equal	Δ	p
`independent`	6.588	15.365	39.488	6.588	**6.479**	12.373	−0.109	0.056
`exchangeable`	**7.289**	19.179	44.870	**7.289**	7.295	14.463	+0.006	0.942
`ar1`	8.496	19.460	45.607	8.540	**8.334**	15.292	−0.206	0.003
`rand_intercept`	8.448	19.412	43.424	8.448	**8.446**	15.507	−0.003	0.967
`nonstationary`	8.049	18.500	44.749	8.049	**7.933**	14.699	−0.116	0.144

E The transitional extension

The simulation behind the estimand discussion of §3 generates a transitional truth, $Y_{i0} = 0$ and $Y_{it} = \rho Y_{i,t-1} + a_t g(\mathbf{X}_{it}) + \varepsilon_{it}$ with $\rho = 0.6$, T=8, n=500, innovation SNR 2, a smooth a_t, and $g \in \{g_{\text{base}}, g_{\text{int}}\}$ (40 replicates). The conditional (transitional) estimand is $c_{it} = \rho Y_{i,t-1} + a_t g(\mathbf{X}_{it})$; the implied marginal estimand is $\mu_{it} = \sum_{s \le t} \rho^{t-s} a_s g(\mathbf{X}_{is})$, a member of class (1) with geometrically

decaying weights. Marginal estimators (DLNM, SMOOTHCUM, GRU; exposure history only) are scored by ISE against μ; transitional estimators (each of the three with $Y_{i,t-1}$ available: a learned-ρ smooth per-visit head, an additive ridge with $Y_{i,t-1}$, a GRU fed $[X_t, Y_{t-1}]$) are scored against c.

Table 16: Transitional-extension simulation (40 replicates): ISE of each estimator against its own estimand (μ for the marginal class, c for the transitional class), $\pm$ Monte Carlo SE.

class	method	additive innovation	non-additive innovation
marginal	DLNM	$\mathbf{0.33 \pm 0.03}$	1.33 ± 0.04
marginal	SMOOTHCUM	0.92 ± 0.03	1.11 ± 0.03
marginal	GRU	0.80 ± 0.04	1.04 ± 0.03
transitional	additive ridge + Y_{t-1}	0.17 ± 0.02	0.83 ± 0.02
transitional	smooth head + $\hat{\rho} Y_{t-1}$	$\mathbf{0.12 \pm 0.01}$	$\mathbf{0.23 \pm 0.01}$
transitional	GRU on $[X_t, Y_{t-1}]$	0.52 ± 0.03	0.65 ± 0.02

Three points complete the estimand discussion of §3. The transitional estimators, which observe Y_{t-1}, predict far better than any marginal estimator (test MSE 0.62–1.32 against 1.52–2.05), as tracking-dominated cohorts lead one to expect; the learned autoregressive coefficient recovers the truth ($\hat{\rho} = 0.606$ and 0.602 against $\rho = 0.6$). Each class recovers its own estimand, so the choice between them is the analyst's question, not a modeling error. And the architecture ordering transfers: on the non-additive innovation the additive restriction costs $3.6\times$ inside the transitional class (0.83 vs. 0.23), the same boundary we measure for the marginal class in §6.3. On the additive innovation the two transitional heads are close (0.12 vs. 0.17, the smooth head slightly ahead); this does not contradict the boundary, because g_{base} is only predominantly additive and the transitional fit is a per-visit problem with $n_B T$ pooled observations, not the cumulative problem whose additive-side bias advantage we quantify in §6.3; the sharp, transferable contrast is the non-additive side.

F COHORT CONSTRUCTION AND FULL COMPARISON

The application uses the ELEMENT birth cohort (Perng et al., 2019): child diet recalls linked to anthropometry, on a dense five-age complete-case window $\{12, 18, 24, 30, 36\}$ months with no imputation. The nutrient panel holds d=17 energy-adjusted (TEI) nutrients—total energy, macronutrients, fiber, sugar, and key minerals and vitamins—for n=665 children; the food-group panel measures the same diets through d=40 energy-adjusted food groups for n=664. The outcome is the WHO weight-for-length z-score at each visit, standardized by training-split statistics. Evaluation uses 20 repeated subject-level 80/20 splits; every tuning choice, the outcome standardization, and the PCA projection (components fit on training subjects only, applied to test subjects) stay inside the training split. Methods and tuning grids match the simulation protocol (§6.1), with the oracle-covariance variant dropped because no true covariance exists.

Full ranking on the nutrient panel (mean test MSE over 20 splits): DLNM 1.033, SMOOTHCUM 1.120, sub-network 1.303, GRU 1.530, followed by the remaining sequence and contemporaneous baselines. The ratios reported in §6.4 follow: best sequence model over SMOOTHCUM 1.37, sub-network over SMOOTHCUM 1.16, DLNM over SMOOTHCUM 0.92. On the food-group panel the raw d=40 arm gives SMOOTHCUM 1.320 against GRU 1.007 (best sequence model, $0.76\times$); the PCA-10 arm gives SMOOTHCUM 1.046 (-21%) with the sequence models essentially unchanged (-0.2 to $+1.9\%$); the PCA-5 arm gives 1.031, confirming the monotone recovery. All food-group arms come from one run set sharing splits and configuration.

The cohort's second exposure panel demonstrates the nuisance-dimension boundary and its remedy. The food-group panel measures the same diets through d=40 correlated components; there SMOOTHCUM's error rises to 1.32 and the best sequence model moves ahead (with 24% lower error), reproducing the erosion with the covariate dimension d of §6.3 on real data. Projecting to 10 principal components recovers most of it: SMOOTHCUM's error falls 21% (to 1.05) while the sequence models move by at most 1.9%, matching the simulation's prediction that the penalty attaches to the per-visit function estimate and is removed by reducing the input dimension.

Two public cohorts with much larger samples confirm the negative prediction of the bias-variance decomposition: once n reaches the thousands there is little variance left to remove, and the advan-

tage vanishes. On HRS cognition (Sonnega et al., 2014) (d=7, T=8, n=2510) every ratio of a competitor's error to that of SMOOTHCUMsits at 0.99–1.01; on ECLS-K reading (Tourangeau et al., 2009) (d=4, T=6, n=6318) the sub-network estimator and the best sequence model are ahead (0.71 and 0.70), so at that scale SMOOTHCUMshould not be used. An advantage that appears where the decomposition predicts it and disappears where the decomposition predicts its absence is evidence for the explanation; both directions are part of the result. Subsampling subjects within HRS confirms the sample-size effect inside one cohort, free of any confounding between cohorts: the advantage over the sub-network estimator grows from 1.01 ± 0.00 at n=2510 to 1.12 ± 0.03 at n=300.

F.1 COHORT CONSTRUCTION

HRS. The Health and Retirement Study (Sonnega et al., 2014), RAND HRS Longitudinal File 2022 (V1) (RAND Center for the Study of Aging, 2025), respondents identified by `hhidpn`. Total cognition (`cogtot`, 0–35) and seven exposures are cleanly coded over waves 3–13; we use the dense eight-wave window **waves 4–11** (≈1998–2012), complete case, no imputation: $n = 2{,}510$ (the lagged-residual panel retains 2,520). Complete-case retention is 5.5% of 45,234 records, cognition is not administered at proxy interviews, and the eight-wave requirement compounds mortality and attrition, so the sample is a survivor cohort. The seven exposures are hypertension ($\bar{x} = 0.55$), diabetes (0.16), heart disease (0.27), stroke (0.085), BMI (26.7), current smoking (0.055) and CES-D depressive symptoms (0–8, mean 1.26). Cognition declines monotonically with age across the window (24.2 at age 70 to 19.5 at age 84), the expected cognitive-aging trajectory.

ELEMENT. The Early Life Exposure in Mexico to ENvironmental Toxicants project (Perng et al., 2019); child diet linked to anthropometry by `proyecto_folio`. We use the dense five-age complete-case window $\{12, 18, 24, 30, 36\}$ months ($T = 5$), no imputation, giving $n \approx 664$–734. Covariates are the 40 TEI-adjusted child diet food groups (results for a curated 17-nutrient covariate set are recorded in the internal project report and are not part of this submission). Binary extremes are defined on the WHO weight-for-length z-score: obesity $z > 1.86$ (q_{95}), undernutrition $z < -1.38$ (q_{10}), severe undernutrition $z < -1.87$ (q_5, ≈ 63 positive cases). For binary outcomes Y is not standardized, so the reported MSE is the Brier score and AUC is the headline metric. Complete-case children are healthier than the pooled sample (undernutrition-10 prevalence 6% against 9.9%).

Time alignment. The food-group `edad` and the anthropometry `stage` codes are clean nominal visit codes; `stage` uses a '2'-prefixed month code (21→1 mo, 212→12 mo), decoded by stripping the leading '2'. The nutrient file differs: uppercase keys, `FOLIO` stored as float, and `EDAD` a *raw* age (12, 13, 14, 18, 19, 20, 24, …), binned to the nearest nominal visit; only 9 of 1,573 children collide, and 99% match the food-group visit set. Complete-case filtering requires *finite* diet vectors, since one stray-NaN row exists.

F.2 FULL METHOD COMPARISONS

We give the complete twelve-method comparison on HRS in Table 17, both on the raw outcome and on the leakage-free tracking residual $r_{it} = Y_{it} - (\hat{\lambda}_{0t} + \hat{\lambda}_{1t} Y_{i,t-1})$ with per-visit one-lag coefficients $\hat{\lambda}_{0t}, \hat{\lambda}_{1t}$ fit on the training subjects. Two patterns matter. *History dominates current status*: every history model attains a test MSE of 0.85–0.87 against 0.95 and 0.98 for the two contemporaneous ones, with the outcome standardized so that 1 is the mean-only predictor, so accumulated burden rather than current status carries the signal; the linear-in-covariates functional-linear model trails at 0.92, locating part of the gain in nonlinear per-covariate effects. *The outcome tracks strongly*: a one-lag autoregression alone brings the test MSE of cognition down to 0.59 (from 0.66 at the first wave to 0.47 at the last), and on the residual every method is within 0.01 of the mean-only predictor; accumulated burden is largely already encoded in prior cognition, though the raw cumulative-history signal is real.

Table 17: HRS cognition, all methods, 20 random $80/20$ subject-level splits. Test MSE (outcome standardized on the training subjects, so $\mathrm{MSE} = 1$ is the mean-only predictor). Ranked by raw-outcome MSE.

Method	raw	tracking residual
DLNM	0.852	0.993
Transformer	0.854	0.992
GRU	0.855	0.993
GBM	0.859	0.993
SMOOTHCUM (proposed)	0.862	0.996
Subnet-Cum	0.869	1.001
TCN	0.897	0.995
TwoStep (Fan)	0.919	1.141
Functional-linear	0.921	0.993
GEE (history)	0.941	0.994
NN (current)	0.947	1.001
GEE (current)	0.976	0.993

F.3 THE SIEVE'S CROSS-VALIDATED K, AND HOW A MIS-TUNED BRANCH FAILS

We select K by subject-level cross-validation inside each training split rather than fixing it. On HRS this leaves the sieve essentially unchanged (Table 18: test MSE 0.862–0.863 against 0.862 at a fixed $K{=}3$), because $K{=}3$ is already near-optimal at $T{=}8$ and there is nothing to recover; adding validation-based early stopping and a width sweep on top changes nothing further. Inside the combination, the cross-validated branch carries 0.21 of the three-family weight on HRS.

The episode is worth recording for *how* a mis-tuned branch fails inside a combination. When we ran the sieve at a hard-coded K on a panel where that K badly under-smooths, its model-averaging weight simply fell towards zero: no error, no warning, and no degradation of the combined prediction large enough to notice; the pool silently reverted to its other members (Table 19). A weight-based combination is therefore *robust* to a broken branch, in that it will not blow up, and *opaque*, in that it will not tell you. Each member must be validated on its own before it is pooled, and a branch whose weight collapses should be read as a diagnostic signal rather than as evidence that its model class is unhelpful.

Table 18: The sieve's cross-validated-K variants on HRS (20 splits). `SmoothCum_Kcv` selects K by subject-level cross-validation; `SmoothCum_tuned` adds a width sweep and validation-based early stopping on top; `SmoothCum` is the fixed-$K{=}3$ run of Table 17.

Cohort	Variant	test MSE
HRS	`SmoothCum_Kcv`	0.8628
HRS	`SmoothCum_tuned`	0.8619
HRS	`SmoothCum` (fixed $K{=}3$)	0.8618

Table 19: Real-data combination with the sieve branch at fixed $K{=}3$ versus at its cross-validated K (20 splits each; test MSE for HRS, lower is better, and test AUC for the ELEMENT extremes, higher is better). CUMBLEND is the three-family pool, $\hat{w}_S$ its sieve weight, *Sel* the CV-selected single family. At fixed K the combined predictor is worse on obesity and severe undernutrition (by 0.011–0.016 AUC) and within Monte-Carlo error on the rest.

	fixed $K{=}3$				cross-validated K			
Task	SMOOTHCUM	CUMBLEND	$\hat{w}_S$	Sel	SMOOTHCUM	CUMBLEND	$\hat{w}_S$	Sel
HRS cognition	0.869	0.856	0.17	0.860	0.870	0.856	0.21	0.861
Obesity	0.540	0.621	0.12	0.570	0.579	0.637	0.07	0.591
Undernut-10	0.542	0.548	0.12	0.523	0.543	0.547	0.07	0.527
Undernut-5	0.492	0.569	0.13	0.588	0.517	0.580	0.11	0.597

G IMPLEMENTATION, HYPERPARAMETERS AND COMPUTE

Neural methods: full training and tuning specification. All neural methods train by full-batch Adam on square loss for 400 epochs (Transformer: 300), and are tuned by the same light uniform protocol: every (width, learning rate) combination is trained on a subject-level $80/20$ split of the training subjects and the combination with the best held-out MSE is refit; learning rates are $\{10^{-3}, 3 \times 10^{-3}\}$ everywhere. The width grids and depths are: contemporaneous MLP $\{16, 32\} \times 3$ hidden layers; the cumulative models (Subnet-Cum and SMOOTHCUM) $\{8, 16\} \times 2$; GRU $\{16, 32\} \times 1$ recurrent layer; Transformer $\{32\} \times 2$ encoder layers with 4 heads; TCN $\{16, 32\} \times 2$ blocks; the Fan–Zhang first-stage network $\{16, 32\} \times 2$. SMOOTHCUM uses the quadratic B-spline basis with $K{=}3$ on the architecture grid and cross-validated K on real data (Appendix F). The sequence models (GRU, Transformer, TCN) additionally use validation-based *early stopping*: an internal 20% subject holdout is tracked every epoch, training stops after 30 epochs without improvement (minimum 40 epochs), and the best-validation checkpoint is restored; each model trains to its own held-out optimum rather than a fixed epoch count, because these unconstrained models otherwise overfit the short, small panels. This is what makes the "structural prior, not capacity" reading of the crossover defensible: a deficit reflects inductive bias, not under-training *or* over-fitting. The Transformer is a causal encoder with sinusoidal position encoding and a causal self-attention mask, so $\widehat{Y}_{it}$ attends only to $s \le t$; the TCN is dilated and left-padded with receptive field growing $1, 2, 4, \ldots$ to span the history.

Structured and classical baselines. DLNM is an additive penalized-spline cross-basis $\sum_j \sum_{s \le t} f_j(X_{ijs}, t - s)$ (Gasparrini et al., 2010), implemented exactly as follows: for each covariate j a cubic B-spline in the covariate value (`SplineTransformer`, 4 knots, fit on the training data), tensored with a polynomial basis of degree < 3 in the lag $t{-}s$ scaled to $[0, 1]$; the cumulative design row at (i, t) sums these tensor features over $s \le t$, and the pooled model (intercept $+\ 1{+}d \cdot n_{xb} \cdot 3$ columns) is fit by ridge regression with $\alpha = 1$. The effect is therefore *lag-indexed*, as the main paper's §3 discusses: the contribution of $\mathbf{X}_{is}$ to a later outcome depends on $t - s$, not on s. The functional-linear cumulative model is its linear-in-covariates counterpart, $\sum_j \sum_{s \le t} \beta_j(t - s) X_{ijs}$, included to isolate cumulative structure from covariate nonlinearity. TwoStep is Fan & Zhang (2000) on first differences. GBM is histogram gradient boosting on the history-expanded design (depth 3, learning rate 0.05, 300 iterations). GEE is linear, in current and history-expanded forms.

Which baselines appear where, and why. The 150-setting comparison grid uses the GRU, the causal Transformer and the TCN as its sequence baselines. Recent long-sequence architectures are not in that grid because their machinery targets long-range dependence over sequences of hundreds to thousands of steps, and at $T \le 12$ they add tuning burden without testing anything the three baselines do not; two of them, a selective state-space model in the Mamba/S4 line and a patch transformer in the PatchTST line, are evaluated in the scale study of Appendix C.4, where the conclusion is unchanged. Continuous-time models are designed for irregularly sampled time; on the regular grids of the simulation studies they offer no advantage over the recurrent and convolutional models, so they are omitted there, and the ODE-RNN is evaluated in the irregular-time study of §6.3, where the SMOOTHCUM time basis, being defined at any real s, applies directly. The neural controlled differential equation is not compared against anywhere in the paper.

Combination. Weights solve a simplex-constrained least-squares problem on out-of-fold predictions, a convex quadratic program over the probability simplex $\Delta_3 = \{w \in \mathbb{R}^3 : w_j \ge 0,\ \sum_j w_j = 1\}$, solved by evaluating a 0.02-spaced grid (1,326 points, one matrix product), which at this dimension is indistinguishable from the exact QP solution. Simulation uses $V = 4$ subject-level folds, real data $V = 3$. Restricted pools support the grid only on the retained coordinates. Weight estimation and family refitting are the only steps between the out-of-fold predictions and the combined predictor, and no test-set quantity enters any of them. Tuning is *nested*: within each of the V model-averaging folds every family's hyperparameter search (width $\times$ learning rate, and on real data the sieve's K) is re-run on that fold's training subjects alone, so no out-of-fold prediction is produced with a hyperparameter chosen using its own fold.

Runtime. Two clocks matter. The neural representations (SMOOTHCUM, Subnet-Cum, GRU, Transformer, TCN, the contemporaneous MLP and the Fan–Zhang first stage) are trained *and tuned once per replicate* in a shared step costing $\approx 40\,\mathrm{s}$ ($39.7 \pm 2.5\,\mathrm{s}$ at $n{=}200$, $40.9 \pm 1.4\,\mathrm{s}$ at $n{=}500$; dominated by the fixed epoch budgets, hence nearly flat in n). The per-method incremental times (SMOOTHCUM $0.06\,\mathrm{s}$, Subnet-Cum $0.15\,\mathrm{s}$, GRU / Transformer / TCN 0.003–$0.005\,\mathrm{s}$, TwoStep $0.14\,\mathrm{s}$) are the *incremental* cost of producing each method's predictions given those trained representations (for the sequence models, essentially a forward pass), not training times; DLNM ($0.93\,\mathrm{s}$), GBM ($0.02\,\mathrm{s}$) and GEE ($0.16\,\mathrm{s}$) are trained per method and their figures are full fits. Method-to-method training-time comparisons should therefore be read from the shared $40\,\mathrm{s}$ step, which all neural methods split, not from the incremental times. The combination adds *no* model fits beyond those the cross-validated selection rule already performs. The full study (150 settings $\times$ 200 replicates for the architecture grid, the K-sweep repeated over all 150 settings, the 60-setting stress slice, 17 settings $\times$ 100 replicates for the combination study plus the 5-point cross-time model-averaging sweep at 100 replicates, and 20 splits $\times$ 4 real tasks) was run as SLURM job arrays on a CPU cluster.

Code layout. Model and estimator code in `smoothcum/` (`methods.py`, `runner.py`, and the loaders `realdata.py`, `realdata_hrs.py`); drivers and aggregation in `scripts/` (`run_hrs.py`, `run_realdata.py`, `model_averaging.py`, `run_model_averaging_cohorts.py`, `additive_share.py`, `aggregate.py`). Every combination-study setting reports 100 Monte-Carlo replicates, every architecture setting 200, and every real-data number 20 subject-level splits, with the split seed shared across methods so that all paired comparisons are on identical partitions. Code and configuration files are available at the repository named in the reproducibility statement.