Globally Aligned KDSUM: A Positive Semidefinite Mixed-Type Kernel for Low-Dimensional Embedding

Kejun Fang
McGill University

May 2026

Abstract

Mixed-type data combine continuous, unordered categorical, and ordered categorical variables, making low-dimensional representation depend on a pairwise geometry that can compare heterogeneous observations and remain valid for spectral or latent-variable embedding. Many mixed-type distances are useful for comparison or clustering, but their double-centered matrices need not be positive semidefinite, limiting their use in principal coordinate analysis, kernel principal component analysis, and fixed-kernel probabilistic principal coordinate analysis. We show that KDSUM, built from Gaussian, Aitken, and Wang–van Ryzin component kernels, defines a positive semidefinite mixed-type sample kernel. Its centered form therefore induces squared Hilbert-space distances and gives an admissible input to these embedding methods. We further propose globally aligned KDSUM (GA-KDSUM), a trace-balanced weighting scheme that favors component kernels aligned with the consensus geometry of the remaining variables while preserving positive semidefiniteness through nonnegative kernel combinations. In latent-factor simulations, GA-KDSUM gives its strongest gains in the setting with many irrelevant variables, outperforming EW-KDSUM and Gower-type PCO baselines, including PSD-corrected Gower PCO. Diagnostics confirm that KDSUM-based kernels are numerically PSD whereas Gower-type double-centered matrices are often indefinite. On two Student Performance prediction tasks, GA-KDSUM achieves the highest mean balanced accuracy and AUC among the compared embeddings based on mixed-type pairwise geometry. The method is therefore best viewed as a PSD kernel interface and embedding strategy for heterogeneous data with shared low-dimensional structure.

Keywords: mixed-type data, positive semidefinite kernels, KDSUM, principal coordinate analysis, kernel alignment, probabilistic embedding

1 Introduction

Modern applications often begin with heterogeneous records rather than purely numerical feature matrices. Electronic health records may contain laboratory measurements, diagnosis and procedure codes, medication indicators, and ordered clinical scores. Educational datasets often combine grades, demographic variables, school attributes, and ordinal behavioral responses. Credit and customer analytics similarly combine continuous financial measurements with unordered and ordered categorical descriptors. In these settings, a low-dimensional representation is useful not only for visualization but also for downstream classification, regression, cohort discovery, segmentation, and model diagnostics (Miotto et al., 2016, Cortez & Silva, 2008, Yeh & Lien, 2009).

The central obstacle is that an embedding method needs a geometry before it can be applied. PCA works with a numerical matrix and is tied to best low-rank approximation (Eckart & Young, 1936, Jolliffe, 2002). Principal coordinate analysis and classical multidimensional scaling start from pairwise dissimilarities, but the dissimilarities must be Euclidean for the centered inner product matrix to be positive semidefinite (Gower, 1966, Borg & Groenen, 1997). Kernel PCA replaces explicit coordinates by a positive semidefinite kernel matrix (Schölkopf et al., 1998, Williams, 2002). Probabilistic principal coordinate analysis gives PCO a latent-variable and maximum-likelihood interpretation, but it is also conditional on a fixed positive semidefinite matrix $\boldsymbol{Q}$ (Zhang & Jordan, 2009). Thus, for mixed-type data, the problem is not only how to compare heterogeneous observations, but how to build a comparison that is admissible as a kernel geometry.

Several approaches are commonly used to compare or model mixed-type observations. Dummy coding embeds categorical variables as indicator vectors, but it treats ordered categories as nominal unless extra structure is imposed and it can let category cardinality affect the geometry (Foss et al., 2019). Gower's coefficient averages range-scaled numerical similarities and matching-based categorical similarities, making it a standard mixed-type baseline (Gower, 1971). Ordinal extensions of Gower replace exact matching on ordered variables by rank-aware comparisons (Podani, 1999). The k-prototypes objective combines squared Euclidean loss for numerical variables with matching loss for categorical variables (Huang, 1998). Semiparametric and latent-variable mixture methods, such as KAMILA and clustMD, model mixed-type clusters through distributional assumptions rather than through a single embedding kernel (Foss et al., 2016, McParland & Gormley, 2016). KDSUM takes a kernel-based route by combining Gaussian, Aitken, and Wang–van Ryzin component kernels into a mixed-type similarity and using bandwidth selection to control the contribution of each variable (Ghashti & Thompson, 2025).

These methods address important parts of the mixed-type problem, but they leave a gap for probabilistic and spectral embedding. A useful mixed-type distance or clustering objective is not automatically a valid positive semidefinite kernel. An indefinite similarity or double-centered distance matrix breaks the Hilbert-space interpretation required by kernel PCA, PCO, and fixed-$\boldsymbol{Q}$ PPCO. Conversely, a positive semidefinite mixed-type kernel immediately induces squared Hilbert-space distances, a centered PCO matrix, and a probabilistic embedding model. This paper focuses on that interface.

We show that the standard KDSUM similarity constructed from Gaussian kernels for continuous variables, Aitken kernels for unordered categorical variables, and Wang–van Ryzin kernels for ordered categorical variables is positive semidefinite. After centering, the resulting matrix

$$\boldsymbol{Q} = \boldsymbol{H}\boldsymbol{\Psi}_{\mathrm{KDSUM}}\boldsymbol{H}$$

is therefore a valid input to PCO, KPCA, and fixed-$\boldsymbol{Q}$ PPCO. The method is conceptually simple. It builds type-specific positive semidefinite component kernels, combines them with nonnegative weights, centers the resulting KDSUM kernel, and applies the PPCO embedding. Since PPCO treats $\boldsymbol{Q}$ as fixed, bandwidths and variable weights are chosen before PPCO is applied rather than by optimizing the PPCO likelihood.

We also introduce GA-KDSUM, a globally aligned weighted KDSUM construction for settings where several heterogeneous variables share a low-dimensional geometry but many additional variables are weakly related or irrelevant. Each component kernel is trace-balanced after centering, and its weight is determined by how strongly its centered geometry aligns with the average centered geometry of the other components. The idea is related to centered kernel alignment, which measures agreement between kernel matrices and has been used for learning convex combinations of kernels (Cortes et al.,

2012). The resulting GA-KDSUM kernel remains positive semidefinite because it is a nonnegative combination of positive semidefinite component kernels.

The proposed embedding is intended for settings where mixed-type observations need to be represented by low-dimensional coordinates. These settings include patient or student representation learning, survey and behavioral data visualization, credit approval and risk screening, customer segmentation, and preprocessing for downstream supervised models. Sections 5 and 6 evaluate this role using synthetic latent factor recovery and held-out downstream prediction. This paper makes three specific contributions:

1. It proves that the standard KDSUM similarity is a positive semidefinite mixed-type kernel under the Gaussian, Aitken, and Wang–van Ryzin construction.
2. It connects KDSUM to PCO, KPCA, and fixed-$\boldsymbol{Q}$ PPCO through the centered kernel matrix $\boldsymbol{Q} = \boldsymbol{H}\boldsymbol{\Psi}\boldsymbol{H}$.
3. It proposes GA-KDSUM, a trace-balanced globally aligned KDSUM embedding that preserves positive semidefiniteness and improves mixed-type pairwise geometry in the setting with many irrelevant variables and in real downstream prediction.

Organization of the paper. Section 2 introduces the mixed-type KDSUM geometry and reviews its connection to PCO, KPCA, and PPCO. Section 3 presents the PSD result, the trace-balanced globally aligned kernel construction, and the embedding algorithm. Section 4 defines the evaluation criteria used in the experiments. Section 5 reports the simulation experiment. Section 6 presents downstream prediction experiments on mixed-type Student Performance datasets. Section 7 discusses interpretation, limitations, and future work, and Section 8 concludes. The appendix gives proof details and supplementary diagnostics.

2 Mixed-Type Kernel Geometry

2.1 Mixed-type data and KDSUM

Consider an $n \times p$ mixed-type data matrix $\boldsymbol{X}$ consisting of n observations and p variables. Assume that the variables are arranged so that the first p_c variables are continuous, the next p_u variables are unordered categorical, and the last p_o variables are ordered categorical, with

$$p = p_c + p_u + p_o.$$

The rows of $\boldsymbol{X}$ are observation vectors $\boldsymbol{x}_i$. Let $\mathbf{1}_n$ denote the n-vector of ones, and let

$$\boldsymbol{\lambda} = \{\boldsymbol{\lambda}^c, \boldsymbol{\lambda}^u, \boldsymbol{\lambda}^o\}$$

denote the collection of variable-specific bandwidths.

For a continuous variable, let

$$k(z) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{z^2}{2}\right),$$

so that the continuous KDSUM contribution for variable k is

$$\frac{1}{\lambda_k} k\left(\frac{x_{ik}-x_{jk}}{\lambda_k}\right), \qquad \lambda_k > 0.$$

For an unordered categorical variable, we use the Aitken kernel (Aitchison & Aitken, 1976),

$$L(a,b,\lambda_k) = \begin{cases} 1, & a = b, \\ \lambda_k, & a \neq b, \end{cases} \qquad \lambda_k \in [0,1].$$

For ordered categorical variables, levels are assumed to be encoded by their integer ranks. We use the Wang and van Ryzin kernel (Wang & Van Ryzin, 1981),

$$l(a,b,\lambda_k) = \begin{cases} 1-\lambda_k, & a = b, \\ \frac{1}{2}(1-\lambda_k)\lambda_k^{|a-b|}, & a \neq b, \end{cases} \qquad \lambda_k \in [0,1].$$

The pairwise KDSUM similarity is

$$\psi(\boldsymbol{x}_i, \boldsymbol{x}_j \mid \boldsymbol{\lambda}) = \prod_{k=1}^{p_c} \frac{1}{\lambda_k} k\left(\frac{x_{ik}-x_{jk}}{\lambda_k}\right) + \sum_{k=p_c+1}^{p_c+p_u} L(x_{ik}, x_{jk}, \lambda_k) + \sum_{k=p_c+p_u+1}^{p} l(x_{ik}, x_{jk}, \lambda_k).$$

When $p_c = 0$, the continuous product is interpreted as the empty product, namely the constant kernel equal to one. At the sample level this contributes the rank-one term $\mathbf{1}_n\mathbf{1}_n^\top$ to the raw similarity. Centering removes this term and it contributes zero to the induced squared distances.

The corresponding KDSUM similarity matrix is denoted by

$$\boldsymbol{\Psi}_{\mathrm{KDSUM}}(\boldsymbol{\lambda}) = \left[\psi(\boldsymbol{x}_i, \boldsymbol{x}_j \mid \boldsymbol{\lambda})\right]_{i,j=1}^{n}.$$

The induced KDSUM dissimilarity is

$$d(\boldsymbol{x}_i, \boldsymbol{x}_j \mid \boldsymbol{\lambda}) = \psi(\boldsymbol{x}_i, \boldsymbol{x}_i \mid \boldsymbol{\lambda}) + \psi(\boldsymbol{x}_j, \boldsymbol{x}_j \mid \boldsymbol{\lambda}) - 2\psi(\boldsymbol{x}_i, \boldsymbol{x}_j \mid \boldsymbol{\lambda}).$$

Although we keep the original KDSUM notation d, in the PCO construction this quantity is used as a squared dissimilarity. Thus we write

$$\delta_{ij}^2 = d(\boldsymbol{x}_i, \boldsymbol{x}_j \mid \boldsymbol{\lambda}).$$

When $\boldsymbol{\Psi}_{\mathrm{KDSUM}}$ is PSD, δ_{ij}^2 is a squared Hilbert-space distance, and δ_{ij} is the associated Hilbert-space distance.

The original KDSUM paper uses this distance mainly for clustering and uses maximum similarity cross-validation (MSCV) to select $\boldsymbol{\lambda}$ (Ghashti & Thompson, 2025). In this paper, MSCV is treated as an external KDSUM bandwidth selection rule. The experiments do not use the PPCO likelihood to tune $\boldsymbol{\lambda}$. The empirical implementation uses a trace-balanced weighted component version of the same PSD construction so that individual variables can be weighted. Accordingly, EW-KDSUM and GA-KDSUM are componentized PSD KDSUM extensions rather than algebraically identical copies of the raw KDSUM aggregation.

2.2 PCO, kernels, and PPCO

Given a squared dissimilarity matrix

$$\boldsymbol{\Delta} = [\delta_{ij}^2],$$

classical PCO forms

$$\boldsymbol{Q} = -\frac{1}{2}\boldsymbol{H}\boldsymbol{\Delta}\boldsymbol{H}, \qquad \boldsymbol{H} = \boldsymbol{I}_n - \frac{1}{n}\mathbf{1}_n\mathbf{1}_n^\top.$$

If $\boldsymbol{Q} \succeq 0$, the dissimilarities are Euclidean. If a positive semidefinite kernel matrix $\boldsymbol{K}$ is available directly, the corresponding centered inner product matrix is

$$\boldsymbol{Q} = \boldsymbol{H}\boldsymbol{K}\boldsymbol{H}.$$

This connects PCO, kernel PCA, and metric multidimensional scaling (Gower, 1966, Williams, 2002, Borg & Groenen, 1997).

Zhang and Jordan's probabilistic PCO (PPCO) model gives PCO a maximum-likelihood interpretation (Zhang & Jordan, 2009). Their model treats the high-dimensional feature vectors as virtual observations and estimates the low-dimensional coordinates conditional on a fixed centered PSD matrix $\boldsymbol{Q}$. Below, the PPCO noise variance is denoted by σ^2_{PPCO} to distinguish it from the KDSUM bandwidth vector $\boldsymbol{\lambda}$.

Let

$$\boldsymbol{Q} = \boldsymbol{U}\boldsymbol{\Gamma}\boldsymbol{U}^\top, \qquad \gamma_1 \geq \cdots \geq \gamma_{n-1} \geq 0,$$

where $\gamma_1, \ldots, \gamma_{n-1}$ are the eigenvalues of the centered matrix $\boldsymbol{Q}$ excluding the trivial eigenvalue associated with $\mathbf{1}_n$, with zero eigenvalues included after the nonzero rank. Let $d_Q = \mathrm{rank}(\boldsymbol{Q})$. For a chosen embedding dimension $1 \leq q < n-1$ with $q \leq d_Q$, the fixed-$\boldsymbol{Q}$ PPCO estimate is

$$\widehat{\boldsymbol{Y}} = \boldsymbol{U}_q(\boldsymbol{\Gamma}_q - \widehat{\sigma}^2_{\mathrm{PPCO}}\boldsymbol{I}_q)^{1/2}\boldsymbol{S},$$

where $\boldsymbol{S}$ is an arbitrary $q \times q$ orthonormal matrix and

$$\widehat{\sigma}^2_{\mathrm{PPCO}} = \frac{1}{n-q-1}\sum_{j=q+1}^{n-1}\gamma_j.$$

Taking $\boldsymbol{S} = \boldsymbol{I}_q$ fixes the rotation. When $\widehat{\sigma}^2_{\mathrm{PPCO}}$ is small, PPCO is nearly the same as classical PCO on the same matrix $\boldsymbol{Q}$. Consequently, any downstream advantage should be interpreted as coming from the mixed-type kernel geometry rather than from PPCO alone.

3 KDSUM to PPCO Embedding

All PSD statements in this section are finite-sample kernel-matrix statements. The bandwidths, trace normalizers, and global-alignment weights are fitted from the training sample and then held fixed for the training kernel matrix and for out-of-sample projection.

3.1 PSD result for the KDSUM similarity

Proposition 1. Let $\boldsymbol{\Psi}_{\mathrm{KDSUM}}$ be the sample KDSUM similarity matrix defined in Section 2.1, using Gaussian kernels for continuous variables, Aitken kernels for unordered categorical variables with bandwidths in $[0, 1]$, and Wang and van Ryzin kernels for ordered categorical variables with bandwidths in $[0, 1]$. Then

$$\boldsymbol{\Psi}_{\mathrm{KDSUM}} \succeq 0.$$

Consequently,

$$\boldsymbol{Q} = \boldsymbol{H}\boldsymbol{\Psi}_{\mathrm{KDSUM}}\boldsymbol{H} \succeq 0.$$

Proof. Each continuous Gaussian component matrix is PSD. The continuous KDSUM block is the Hadamard product of these component matrices, so it is PSD by the Schur product theorem (Horn & Johnson, 2012). If $p_c = 0$, the empty product convention gives the constant kernel $\mathbf{1}_n\mathbf{1}_n^\top$, which is PSD. Omitting this block changes the raw similarity by only a constant rank-one term. This term is removed by centering and contributes zero to the induced squared distances.

For an unordered categorical variable with one-hot category matrix $\boldsymbol{C}$, the Aitken component matrix can be written as

$$\boldsymbol{K}^{(u)} = \lambda\mathbf{1}_n\mathbf{1}_n^\top + (1-\lambda)\boldsymbol{C}\boldsymbol{C}^\top, \qquad \lambda \in [0, 1].$$

This is a nonnegative linear combination of PSD matrices, so $\boldsymbol{K}^{(u)} \succeq 0$.

For an ordered categorical variable with ordered levels $1, \ldots, G$, let $\boldsymbol{C}$ be the one-hot level matrix and define

$$(\boldsymbol{T}_\lambda)_{rs} = \lambda^{|r-s|}, \qquad r, s \in \{1, \ldots, G\}.$$

For $\lambda \in [0, 1]$, $\boldsymbol{T}_\lambda \succeq 0$. Hence $\boldsymbol{A} = \boldsymbol{C}\boldsymbol{T}_\lambda\boldsymbol{C}^\top \succeq 0$. The Wang and van Ryzin component matrix can be written as

$$\boldsymbol{K}^{(o)} = \frac{1-\lambda}{2}\boldsymbol{A} + \frac{1-\lambda}{2}\boldsymbol{C}\boldsymbol{C}^\top.$$

It follows that $\boldsymbol{K}^{(o)} \succeq 0$, since it is a nonnegative linear combination of PSD matrices.

The KDSUM similarity matrix is the sum of the continuous block, the unordered categorical components, and the ordered categorical components. Since each summand is PSD, $\boldsymbol{\Psi}_{\mathrm{KDSUM}} \succeq 0$. Finally, for any $\boldsymbol{v} \in \mathbb{R}^n$,

$$\boldsymbol{v}^\top\boldsymbol{H}\boldsymbol{\Psi}_{\mathrm{KDSUM}}\boldsymbol{H}\boldsymbol{v} = (\boldsymbol{H}\boldsymbol{v})^\top\boldsymbol{\Psi}_{\mathrm{KDSUM}}(\boldsymbol{H}\boldsymbol{v}) \geq 0.$$

Thus, $\boldsymbol{Q} = \boldsymbol{H}\boldsymbol{\Psi}_{\mathrm{KDSUM}}\boldsymbol{H} \succeq 0$.

The standard PSD facts used above are recalled in Appendix A.1. □

Corollary 1. Define

$$\delta_{ij}^2 = (\boldsymbol{\Psi}_{\mathrm{KDSUM}})_{ii} + (\boldsymbol{\Psi}_{\mathrm{KDSUM}})_{jj} - 2(\boldsymbol{\Psi}_{\mathrm{KDSUM}})_{ij}, \qquad \boldsymbol{\Delta} = [\delta_{ij}^2].$$

Then δ_{ij}^2 is a squared Hilbert space distance and

$$-\frac{1}{2}\boldsymbol{H}\boldsymbol{\Delta}\boldsymbol{H} = \boldsymbol{H}\boldsymbol{\Psi}_{\mathrm{KDSUM}}\boldsymbol{H}.$$

The proof is given in Appendix A.2.

3.2 Trace-balanced component KDSUM kernel

Raw component kernels may have different numerical scales. To prevent scale from determining the embedding, we normalize each PSD component kernel by its centered trace before combining the components. Let $\boldsymbol{K}_1, \ldots, \boldsymbol{K}_M$ denote the PSD component kernels used in the embedding construction. In the implementation below, EW-KDSUM and GA-KDSUM use these variable-level components so that weights can be assigned while preserving PSD closure. For each component, define

$$\widetilde{\boldsymbol{K}}_m = \frac{\boldsymbol{K}_m}{\operatorname{tr}(\boldsymbol{H}\boldsymbol{K}_m\boldsymbol{H}) + \epsilon_{\mathrm{tr}}}, \qquad \epsilon_{\mathrm{tr}} > 0.$$

The equal-weight trace-balanced KDSUM baseline is denoted by EW-KDSUM and uses the kernel

$$\boldsymbol{\Psi}_{\mathrm{eq}} = \frac{1}{M} \sum_{m=1}^{M} \widetilde{\boldsymbol{K}}_m.$$

Since each $\widetilde{\boldsymbol{K}}_m$ is a positive scalar multiple of a PSD matrix, it is PSD. Hence

$$\boldsymbol{\Psi}_{\mathrm{eq}} \succeq 0.$$

Remark (Variable-level component construction)**.** In the variable-level implementation, the components $\boldsymbol{K}_1, \ldots, \boldsymbol{K}_M$ are the continuous Gaussian component kernels, the unordered Aitken component kernels, and the ordered Wang and van Ryzin component kernels. The raw KDSUM kernel uses a single continuous product block, whereas the variable-level construction keeps continuous components separate so that each variable can receive its own weight. Both constructions are PSD by the same closure properties.

3.3 Globally aligned weighted extension

Equal weighting can obscure informative components when many irrelevant variables are present. A component that reflects the shared mixed-type geometry should have a centered kernel that agrees with the geometry induced by other components. GA-KDSUM uses this idea directly: it gives a larger weight to a component when its centered trace-balanced kernel aligns with the average centered geometry of the remaining components.

For each trace-balanced component kernel, define

$$\boldsymbol{Q}_m = \boldsymbol{H}\widetilde{\boldsymbol{K}}_m\boldsymbol{H}.$$

For $M > 1$, define the leave-one-component-out global reference geometry

$$\boldsymbol{Q}_{-m} = \frac{1}{M-1} \sum_{r \neq m} \boldsymbol{Q}_r.$$

The global alignment score is

$$a_m = \max\left\{0, \frac{\langle \boldsymbol{Q}_m, \boldsymbol{Q}_{-m} \rangle_F}{\|\boldsymbol{Q}_m\|_F \|\boldsymbol{Q}_{-m}\|_F + \epsilon_{\mathrm{align}}}\right\}, \qquad \epsilon_{\mathrm{align}} > 0.$$

The truncation at zero is used only for numerical stability. The final weight is

$$w_m = (1 - \alpha_w)\frac{1}{M} + \alpha_w \frac{\exp(a_m/\tau_w)}{\sum_{\ell=1}^{M} \exp(a_\ell/\tau_w)}, \qquad \alpha_w \in [0,1], \quad \tau_w > 0.$$

The experiments use the fixed defaults $\alpha_w = 0.70$ and $\tau_w = 0.05$. These values are chosen a priori, kept fixed across all reported simulation and real-data tasks, and not tuned using outcome labels. When $M = 1$, the global reference is undefined, and we set $w_1 = 1$.

Remark (Interpretation of globally aligned weights)**.** The weighting rule emphasizes components that agree with the shared geometry represented across the mixed-type variables. It is not a supervised feature selector and it does not use the outcome labels or the true latent factors. If an irrelevant group of variables shares a strong nuisance geometry, a purely unsupervised global alignment rule can also emphasize that structure. The method is therefore best interpreted as an unsupervised geometry-alignment rule rather than as hard signal selection.

The GA-KDSUM kernel is

$$\boldsymbol{\Psi}_{\mathrm{GA}} = \sum_{m=1}^{M} w_m \widetilde{\boldsymbol{K}}_m.$$

Here $w_m \geq 0$ and $\sum_{m=1}^{M} w_m = 1$.

Corollary 2. The trace-balanced EW-KDSUM kernel $\boldsymbol{\Psi}_{\mathrm{eq}}$ and the GA-KDSUM kernel $\boldsymbol{\Psi}_{\mathrm{GA}}$ are PSD. Consequently, $\boldsymbol{H}\boldsymbol{\Psi}_{\mathrm{eq}}\boldsymbol{H} \succeq 0$ and $\boldsymbol{H}\boldsymbol{\Psi}_{\mathrm{GA}}\boldsymbol{H} \succeq 0$.

Proof. For each component m, $\boldsymbol{K}_m \succeq 0$. Since $\boldsymbol{H}$ is symmetric,

$$\boldsymbol{v}^\top \boldsymbol{H}\boldsymbol{K}_m\boldsymbol{H}\boldsymbol{v} = (\boldsymbol{H}\boldsymbol{v})^\top \boldsymbol{K}_m(\boldsymbol{H}\boldsymbol{v}) \geq 0$$

for every $\boldsymbol{v} \in \mathbb{R}^n$. Hence $\boldsymbol{H}\boldsymbol{K}_m\boldsymbol{H} \succeq 0$, and in particular

$$\mathrm{tr}(\boldsymbol{H}\boldsymbol{K}_m\boldsymbol{H}) \geq 0.$$

Because $\epsilon_{\mathrm{tr}} > 0$, the trace-normalization constant

$$c_m = \{\mathrm{tr}(\boldsymbol{H}\boldsymbol{K}_m\boldsymbol{H}) + \epsilon_{\mathrm{tr}}\}^{-1}$$

is strictly positive. Therefore

$$\widetilde{\boldsymbol{K}}_m = c_m \boldsymbol{K}_m \succeq 0.$$

The EW-KDSUM kernel $\boldsymbol{\Psi}_{\mathrm{eq}}$ is a nonnegative linear combination of the matrices $\widetilde{\boldsymbol{K}}_m$, and is therefore PSD. Similarly, the globally aligned weights are nonnegative by construction and, conditional on the observed data, are fixed scalars summing to one. Thus

$$\boldsymbol{\Psi}_{\mathrm{GA}} = \sum_{m=1}^{M} w_m \widetilde{\boldsymbol{K}}_m$$

is a nonnegative linear combination of PSD matrices, and hence $\boldsymbol{\Psi}_{\mathrm{GA}} \succeq 0$. Finally, centering preserves positive semidefiniteness in the sense that, for any PSD matrix $\boldsymbol{A}$,

$$\boldsymbol{v}^\top \boldsymbol{H}\boldsymbol{A}\boldsymbol{H}\boldsymbol{v} = (\boldsymbol{H}\boldsymbol{v})^\top \boldsymbol{A}(\boldsymbol{H}\boldsymbol{v}) \geq 0$$

for all $\boldsymbol{v} \in \mathbb{R}^n$. Applying this with $\boldsymbol{A} = \boldsymbol{\Psi}_{\mathrm{eq}}$ and $\boldsymbol{A} = \boldsymbol{\Psi}_{\mathrm{GA}}$ gives

$$\boldsymbol{H}\boldsymbol{\Psi}_{\mathrm{eq}}\boldsymbol{H} \succeq 0, \qquad \boldsymbol{H}\boldsymbol{\Psi}_{\mathrm{GA}}\boldsymbol{H} \succeq 0.$$

□

3.4 Algorithm

Algorithm 1 KDSUM-based embedding with equal or global-alignment weighting

Require: Mixed-type data $\boldsymbol{X} = (\boldsymbol{X}_c, \boldsymbol{X}_u, \boldsymbol{X}_o)$, target dimension $1 \le q < n-1$, component bandwidths, and weighting mode

Ensure: Embedding $\widehat{\boldsymbol{Y}}$, centered kernel $\boldsymbol{Q}$, and PSD diagnostics

Build PSD component kernels $\boldsymbol{K}_1, \ldots, \boldsymbol{K}_M$ using Gaussian kernels for continuous variables, Aitken kernels for unordered categorical variables, and Wang and van Ryzin kernels for ordered categorical variables.

For each component, compute $\operatorname{tr}(\boldsymbol{H}\boldsymbol{K}_m\boldsymbol{H})$ and set $\widetilde{\boldsymbol{K}}_m = \boldsymbol{K}_m/(\operatorname{tr}(\boldsymbol{H}\boldsymbol{K}_m\boldsymbol{H}) + \epsilon_{\mathrm{tr}})$.

if $M = 1$ **then**

    Set $w_1 = 1$.

else if the weighting mode is equal **then**

    Set $w_m = 1/M$ for all m.

else

    Compute $\boldsymbol{Q}_m = \boldsymbol{H}\widetilde{\boldsymbol{K}}_m\boldsymbol{H}$, the global reference matrices $\boldsymbol{Q}_{-m}$, the global alignment scores a_m, and weights w_m.

end if

Form $\boldsymbol{\Psi} = \sum_{m=1}^{M} w_m \widetilde{\boldsymbol{K}}_m$.

Center the kernel as $\boldsymbol{Q} = \boldsymbol{H}\boldsymbol{\Psi}\boldsymbol{H}$.

Record PSD diagnostics including symmetry error, centering error, minimum eigenvalue, and the number of eigenvalues below -10^{-8}.

Compute the eigendecomposition $\boldsymbol{Q} = \boldsymbol{U}\boldsymbol{\Gamma}\boldsymbol{U}^\top$, with eigenvalues sorted in decreasing order.

Let $\tilde{\gamma}_j = \max\{\gamma_j, 0\}$ and estimate the PPCO noise variance

$$\widehat{\sigma}^2_{\mathrm{PPCO}} = \frac{1}{n-q-1} \sum_{j=q+1}^{n-1} \tilde{\gamma}_j.$$

Return $\widehat{\boldsymbol{Y}} = \boldsymbol{U}_q(\boldsymbol{\Gamma}_q - \widehat{\sigma}^2_{\mathrm{PPCO}}\boldsymbol{I}_q)_+^{1/2}$, where $(\cdot)_+$ truncates negative diagonal entries to zero.

The component bandwidths are fixed before PPCO is applied. They may be chosen by an external KDSUM rule, such as MSCV, or by a prespecified experimental rule. They are not optimized by the PPCO likelihood. The eigenvalue truncation used to compute $\tilde{\gamma}_j$ is a numerical safeguard for roundoff. Under the PSD result above, all nontrivial eigenvalues γ_j are nonnegative in exact arithmetic. The final $(\cdot)_+$ operation similarly prevents numerical negative entries after subtracting the PPCO noise estimate. Algorithm 1 describes the training embedding. For held-out observations, the bandwidths, trace normalizers, and weights are fitted on the training set and then kept fixed.

For an out-of-sample point $\boldsymbol{x}_*$, let $\boldsymbol{k}_* \in \mathbb{R}^n$ contain its weighted KDSUM kernel similarities to the n training observations. Let $\boldsymbol{\Psi}$ be the training kernel before centering. The centered train–test kernel vector is

$$\boldsymbol{q}_* = \boldsymbol{k}_* - \frac{1}{n}\boldsymbol{\Psi}\mathbf{1}_n - \frac{1}{n}\mathbf{1}_n\mathbf{1}_n^\top\boldsymbol{k}_* + \frac{1}{n^2}\mathbf{1}_n\mathbf{1}_n^\top\boldsymbol{\Psi}\mathbf{1}_n.$$

Given the training eigendecomposition $\boldsymbol{Q} = \boldsymbol{U}\boldsymbol{\Gamma}\boldsymbol{U}^\top$, the out-of-sample PPCO coordinate is

$$\widehat{\boldsymbol{y}}_*^\top = \boldsymbol{q}_*^\top\boldsymbol{U}_q\boldsymbol{\Gamma}_q^+(\boldsymbol{\Gamma}_q - \widehat{\sigma}^2_{\mathrm{PPCO}}\boldsymbol{I}_q)_+^{1/2},$$

where $\boldsymbol{\Gamma}_q^+$ is the Moore–Penrose inverse on the retained eigenspace.

4 Evaluation Criteria

This section defines the evaluation criteria used in Sections 5 and 6. These metrics are not part of the kernel construction. They are collected here to make the comparisons unambiguous.

Latent factor R^2. For synthetic latent factor recovery, let $\boldsymbol{Z} \in \mathbb{R}^{n \times q_z}$ be the true latent factor matrix and let $\boldsymbol{Y} \in \mathbb{R}^{n \times r}$ be an estimated embedding. We first fit the least-squares regression

$$\widehat{\boldsymbol{Z}} = \mathbf{1}_n \widehat{\boldsymbol{\alpha}}^\top + \boldsymbol{Y}\widehat{\boldsymbol{B}}.$$

The latent recovery coefficient is

$$R^2_{\text{latent}} = 1 - \frac{\|\boldsymbol{Z} - \widehat{\boldsymbol{Z}}\|_F^2}{\|\boldsymbol{Z} - \mathbf{1}_n \bar{\boldsymbol{z}}^\top\|_F^2}, \qquad \bar{\boldsymbol{z}} = \frac{1}{n}\boldsymbol{Z}^\top \mathbf{1}_n.$$

Larger values indicate better recovery of the latent factors up to a linear transformation.

PSD diagnostics. For PSD diagnostics, the experiments record

$$\lambda_{\min}(\boldsymbol{Q}), \qquad N_-(\boldsymbol{Q}) = \sum_{j=1}^{n} \mathbf{1}\{\lambda_j(\boldsymbol{Q}) < -10^{-8}\},$$

along with the centering and symmetry errors

$$E_{\text{center}} = \|\boldsymbol{Q}\mathbf{1}_n\|_\infty, \qquad E_{\text{sym}} = \|\boldsymbol{Q} - \boldsymbol{Q}^\top\|_\infty.$$

These diagnostics separate the kernel validity claim from downstream predictive performance.

Paired improvement and win rate. For a score s evaluated over Monte Carlo replicates or cross-validation folds indexed by $b = 1, \ldots, B$, paired improvement over a baseline is

$$\Delta_m = \frac{1}{B}\sum_{b=1}^{B} (s_{mb} - s_{0b}),$$

where s_{mb} is the score of method m and s_{0b} is the score of the baseline on the same split or replicate. In the real-data tables, the default baseline s_{0b} is standard Gower PCO unless another baseline is named explicitly. The win rate is

$$\text{WinRate}_m = \frac{1}{B}\sum_{b=1}^{B} \mathbf{1}\{s_{mb} > s_{0b}\}.$$

The paired comparison is the primary comparative summary because it removes replicate-to-replicate or fold-to-fold variation shared by the methods.

Signal-weight diagnostic. In simulations where the informative variables are known, let $\mathcal{S}$ denote the signal components and let $\mathcal{N}$ denote the noise components. For component weights w_m, define

$$W_{\mathcal{S}} = \sum_{m \in \mathcal{S}} w_m, \qquad W_{\mathcal{N}} = \sum_{m \in \mathcal{N}} w_m.$$

This diagnostic is relevant for the weighted KDSUM construction because the method is designed to emphasize components sharing a common latent geometry. In the simulation experiment, signal

components include all continuous, unordered, and ordered variables generated from the latent factors.

Balanced accuracy. For classification tasks, the primary metric is balanced accuracy. Let $\mathcal{C}_{\mathcal{T}}$ be the set of classes represented in the test fold, and let $C_{\mathcal{T}} = |\mathcal{C}_{\mathcal{T}}|$. If TP_c and FN_c denote the one-vs-rest true positives and false negatives for class c, then

$$\mathrm{BalAcc} = \frac{1}{C_{\mathcal{T}}} \sum_{c \in \mathcal{C}_{\mathcal{T}}} \frac{\mathrm{TP}_c}{\mathrm{TP}_c + \mathrm{FN}_c}.$$

Macro one-vs-rest AUC. For macro one-vs-rest AUC, let $\mathcal{C}_{\mathrm{AUC}}$ be the classes for which the test fold contains at least one positive and one negative observation. Let $n_c^+ = |\{i : y_i = c\}|$ and $n_c^- = |\{j : y_j \neq c\}|$ within the test fold. For the downstream experiments, the class scores s_{ic} are the logistic-regression predicted probabilities for class c. Then

$$\mathrm{MacroAUC} = \frac{1}{|\mathcal{C}_{\mathrm{AUC}}|} \sum_{c \in \mathcal{C}_{\mathrm{AUC}}} \mathrm{AUC}_c,$$

where

$$\mathrm{AUC}_c = \frac{1}{n_c^+ n_c^-} \sum_{i:y_i=c} \sum_{j:y_j \neq c} \left[\mathbf{1}\{s_{ic} > s_{jc}\} + \frac{1}{2} \mathbf{1}\{s_{ic} = s_{jc}\} \right].$$

Undefined class-wise AUC values are omitted from the macro average. If $\mathcal{C}_{\mathrm{AUC}}$ is empty, macro AUC is not reported for that fold.

5 Simulation Experiment

The simulation experiment is designed to answer three questions. First, does the proposed KDSUM construction produce numerically PSD centered kernels? Second, in a setting with many irrelevant mixed-type variables, does the globally aligned KDSUM geometry recover latent factors better than Gower-type pairwise geometries and EW-KDSUM? Third, does the global-alignment weighting rule increase the relative weight assigned to variables generated from the shared latent geometry?

5.1 Data generating process and compared methods

Each replicate used $n = 180$ observations and two latent factors

$$\boldsymbol{z}_i \sim N(0, \boldsymbol{I}_2).$$

The simulation used a mixed-type shared-signal design. The latent factors generated signal variables of all three types. Four continuous signal variables were noisy linear projections of z_1, z_2, $z_1 + z_2$, and $z_1 - z_2$. Six unordered categorical signal variables were generated from latent quadrants, latent directions, nearest latent prototypes, and nonlinear partitions of the latent plane. Four ordered categorical signal variables were generated by discretizing noisy latent projections into seven ordered levels. Additional irrelevant variables were split across continuous, unordered categorical, and ordered categorical types.

The four scenarios were `mixed_clean`, `mixed_noise`, `mixed_high_noise`, and `mixed_many_noise`. The clean scenario contains only the mixed-type signal variables. The noisy scenarios add 24, 48,

and 120 irrelevant mixed-type variables, respectively, with higher observation noise in the latter two settings. Each scenario used $B = 100$ Monte Carlo repetitions. The embedding dimension was fixed at $q = 2$, matching the true latent factor dimension.

The main comparisons are restricted to unsupervised mixed-type pairwise geometries, since the contribution of the paper is a PSD kernel interface for PCO, KPCA, and fixed-$\boldsymbol{Q}$ PPCO. The compared methods were GA-KDSUM, the equal-weight trace-balanced baseline EW-KDSUM, standard Gower PCO, ordinal-aware Gower PCO, and PSD-corrected Gower PCO. Standard Gower treats categorical variables by exact matching. The ordinal-aware Gower baseline treats ordered variables by scaled rank distance. The PSD-corrected Gower baseline repairs the Gower centered matrix by spectral clipping. Specifically, after forming $\boldsymbol{Q}_G = -\frac{1}{2}\boldsymbol{H}\boldsymbol{\Delta}_G\boldsymbol{H}$ from the squared Gower dissimilarity matrix $\boldsymbol{\Delta}_G$, negative eigenvalues are set to zero and PCO is applied to the resulting PSD matrix. This baseline tests whether generic PSD repair is enough to replace a native PSD mixed-type kernel.

5.2 Mixed-type simulations with irrelevant variables

The most informative simulation setting for GA-KDSUM is the setting with many irrelevant mixed-type variables. Table 1 reports latent factor recovery in `mixed_many_noise`. GA-KDSUM achieved the highest mean R^2_{latent}, with a mean of 0.821. The next best method was ordinal Gower PCO with 0.781, followed by EW-KDSUM with 0.769. Standard Gower PCO and PSD-corrected Gower PCO both had mean R^2_{latent} of 0.736.

Table 1: Latent factor recovery in `mixed_many_noise`. Larger R^2_{latent} is better.

Method	Mean R^2_{latent}	SE	Rank
GA-KDSUM	0.821	0.002	1
ordinal Gower PCO	0.781	0.002	2
EW-KDSUM	0.769	0.002	3
Gower PCO	0.736	0.003	4
PSD-corrected Gower PCO	0.736	0.003	4

The paired comparison in Table 2 gives the strongest summary of the result for the setting with many irrelevant variables. On the same Monte Carlo replicates, GA-KDSUM improved over EW-KDSUM by 0.051, over standard Gower PCO by 0.085, over ordinal Gower PCO by 0.039, and over PSD-corrected Gower PCO by 0.085. The win rate was 1.00 against every baseline.

Table 2: Paired advantage of GA-KDSUM in `mixed_many_noise`. Positive values favor GA-KDSUM.

Baseline	GA-KDSUM − baseline	SE	Win rate
EW-KDSUM	0.051	0.001	1.00
Gower PCO	0.085	0.002	1.00
ordinal Gower PCO	0.039	0.001	1.00
PSD-corrected Gower PCO	0.085	0.002	1.00

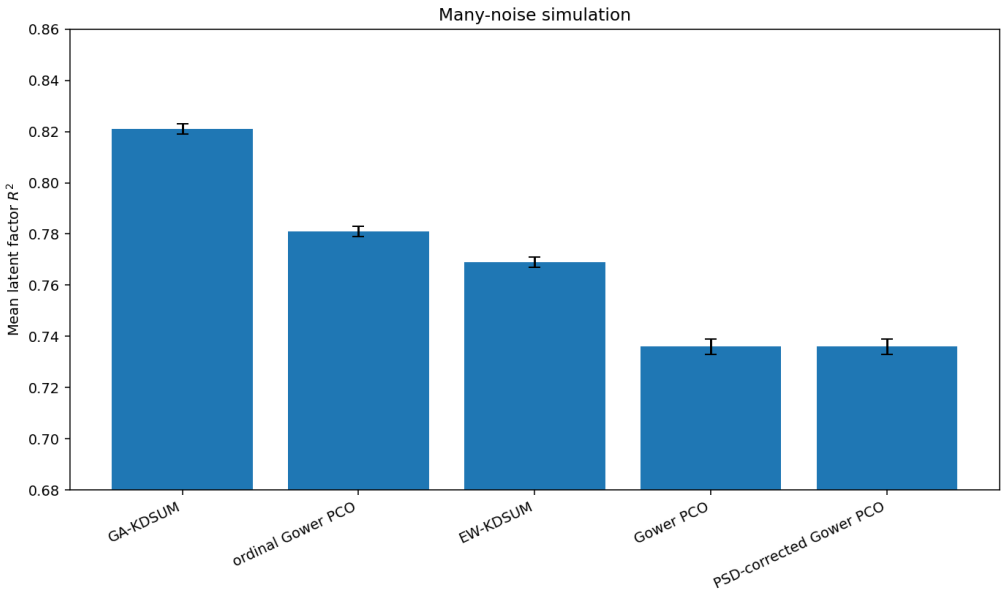

Figure 1: Latent factor recovery in the setting with many irrelevant variables. Error bars show Monte Carlo standard errors.

The full simulation results, reported in Appendix B, show the boundary of the method. In the clean and mild-noise settings, ordinal Gower PCO remains competitive and often gives the highest mean latent recovery. These results suggest that GA-KDSUM is most useful when the data contain a shared latent structure together with many irrelevant mixed-type variables. In this setting, equal weighting can allow unrelated variables to obscure the latent geometry, while the alignment weights emphasize variables that agree with the common pairwise structure and reduce the influence of unrelated variables.

5.3 PSD diagnostics and weighting mechanism

Table 3 summarizes the PSD diagnostics across all simulation replicates. The KDSUM-based methods had no centered-kernel eigenvalues below -10^{-8}. Their minimum eigenvalues were at numerical roundoff scale. In contrast, the double-centered matrices induced by standard Gower PCO and ordinal Gower PCO were often indefinite. Standard Gower PCO had a minimum eigenvalue of -8.22×10^{-1} and 32,833 total negative eigenvalues across the simulation panel. Ordinal Gower PCO had a minimum eigenvalue of -7.52×10^{-1} and 36,080 total negative eigenvalues. The PSD-corrected Gower baseline removed negative eigenvalues but did not improve latent recovery enough to match GA-KDSUM in the setting with many irrelevant variables.

Table 3: PSD diagnostics across all simulation replicates. N_- counts eigenvalues below -10^{-8}.

Method	Min $\lambda_{\min}(\boldsymbol{Q})$	Max N_-	Total N_-	Max E_{center}
GA-KDSUM	-4.09e-16	0	0	1.26e-15
EW-KDSUM	-3.11e-16	0	0	1.14e-15
Gower PCO	-8.22e-01	130	32833	3.43e-14
ordinal Gower PCO	-7.52e-01	123	36080	2.09e-14
PSD-corrected Gower PCO	-3.11e-14	0	0	1.60e-13

The weight diagnostics in Appendix B show that global alignment also changes the contribution of signal components. In `mixed_noise`, GA-KDSUM assigned mean signal weight 0.805, compared with 0.368 under EW-KDSUM. In `mixed_high_noise`, the corresponding weights were 0.615 and 0.226. In `mixed_many_noise`, they were 0.256 and 0.104. These values should be interpreted as evidence of geometry alignment rather than hard feature selection: GA-KDSUM increases the relative contribution of signal components, but its main objective is to align component geometries, not to recover a sparse set of true variables.

6 Downstream Prediction on Student Performance Data

The real-data experiments evaluate whether the proposed PSD mixed-type geometry improves held-out downstream prediction, rather than only recovering latent factors in simulation. The benchmark uses the UCI Student Performance data, which contains two course files with the same variable design: Portuguese and Mathematics (Cortez, 2008, Cortez & Silva, 2008). Both tasks combine continuous, unordered categorical, and ordered categorical variables describing school, demographic, family, and behavioral attributes. The target is pass or fail, defined as $1(G3 \geq 10)$. The previous grade variables $G1$ and $G2$ are excluded to avoid a near-deterministic grade proxy.

6.1 Dataset panel and inductive protocol

Table 4 summarizes the real-data panel. Because all compared embedding methods require pairwise matrices, each course file is capped at a deterministic benchmark subsample of at most 300 observations.

Table 4: Real-data benchmark panel. Both datasets are mixed-type educational classification tasks.

Dataset	n	p_c	p_u	p_o	Role
Student-Portuguese	300	2	17	11	mixed educational classification
Student-Math	300	2	17	11	same-source validation task

The evaluation is inductive. All preprocessing steps, trace normalizers, KDSUM weights, and embeddings are fitted within each training fold. Test observations are embedded only by out-of-sample projection, using the kernel out-of-sample formula for KDSUM methods and the classical MDS out-of-sample formula for Gower PCO. The same PSD repair for PSD-corrected Gower PCO is applied within each training fold. A logistic regression classifier is then fit on the training embedding

and evaluated on the held-out test fold. The reported results use five-fold cross-validation repeated five times, giving 25 paired train-test splits. The embedding dimension was fixed in advance at $q = 5$ for all methods and all splits.

6.2 Downstream performance

Table 5 reports downstream performance. Balanced accuracy is the primary score because pass or fail labels may be imbalanced. AUC is reported as a secondary threshold-free measure. The paired difference and win rate are computed relative to standard Gower PCO on the same cross-validation splits. GA-KDSUM achieved the highest mean balanced accuracy on both tasks. On Student-Portuguese, GA-KDSUM achieved balanced accuracy 0.703, compared with 0.677 for EW-KDSUM, 0.645 for PSD-corrected Gower PCO, 0.639 for standard Gower PCO, and 0.621 for ordinal Gower PCO. On Student-Math, GA-KDSUM achieved balanced accuracy 0.661, compared with 0.605 for EW-KDSUM, 0.578 for ordinal Gower PCO, 0.555 for PSD-corrected Gower PCO, and 0.553 for standard Gower PCO.

Table 5: Real-data downstream performance under repeated five-fold inductive cross-validation. Positive Δ values favor the listed method over standard Gower PCO.

Dataset	Method	Bal. acc.	Δ vs. Gower	Win rate	AUC	Rank
Student-Portuguese	GA-KDSUM	0.703 (0.014)	0.064 (0.015)	0.84	0.735 (0.017)	1
Student-Portuguese	EW-KDSUM	0.677 (0.015)	0.038 (0.014)	0.68	0.734 (0.018)	2
Student-Portuguese	Gower PCO	0.639 (0.015)	–	–	0.679 (0.019)	4
Student-Portuguese	ordinal Gower PCO	0.621 (0.016)	-0.019 (0.012)	0.32	0.657 (0.017)	5
Student-Portuguese	PSD-corrected Gower PCO	0.645 (0.015)	0.006 (0.001)	0.48	0.678 (0.019)	3
Student-Math	GA-KDSUM	0.661 (0.012)	0.108 (0.010)	0.92	0.699 (0.013)	1
Student-Math	EW-KDSUM	0.605 (0.011)	0.052 (0.008)	0.84	0.657 (0.013)	2
Student-Math	Gower PCO	0.553 (0.014)	–	–	0.600 (0.014)	5
Student-Math	ordinal Gower PCO	0.578 (0.013)	0.025 (0.007)	0.76	0.607 (0.014)	3
Student-Math	PSD-corrected Gower PCO	0.555 (0.014)	0.001 (0.001)	0.12	0.600 (0.014)	4

The paired advantages of GA-KDSUM are summarized in Table 6. On Student-Portuguese, GA-KDSUM improved over standard Gower PCO by 0.064 in balanced accuracy, with win rate 0.84. On Student-Math, the paired improvement over Gower PCO was 0.108, with win rate 0.92. The improvement over EW-KDSUM was also positive on both tasks, with 0.026 on Student-Portuguese and 0.056 on Student-Math. These results show that the benefits of GA-KDSUM are not limited to synthetic latent-factor recovery. Under a fully inductive evaluation protocol, the proposed PSD mixed-type geometry produces embeddings that improve held-out prediction relative to both Gower-type PCO baselines and the equal-weight KDSUM baseline.

Table 6: Paired balanced-accuracy advantage of GA-KDSUM over each real-data baseline. Positive values favor GA-KDSUM.

Dataset	Baseline	GA-KDSUM − baseline	SE	Win rate
Student-Math	EW-KDSUM	0.056	0.010	0.88
Student-Math	Gower PCO	0.108	0.010	0.92
Student-Math	ordinal Gower PCO	0.083	0.013	0.88
Student-Math	PSD-corrected Gower PCO	0.107	0.011	0.92
Student-Portuguese	EW-KDSUM	0.026	0.010	0.60
Student-Portuguese	Gower PCO	0.064	0.015	0.84
Student-Portuguese	ordinal Gower PCO	0.082	0.017	0.80
Student-Portuguese	PSD-corrected Gower PCO	0.058	0.015	0.84

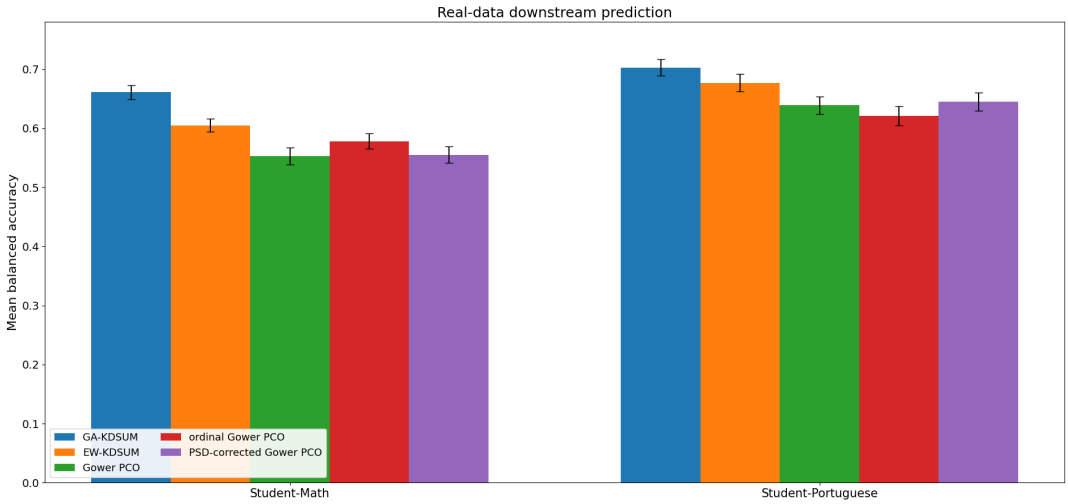

Figure 2: Mean balanced accuracy on the two Student Performance tasks. Error bars show standard errors across repeated cross-validation splits.

As a secondary metric, AUC gives the same broad ordering. GA-KDSUM had the highest mean AUC on Student-Math (0.699) and the highest mean AUC on Student-Portuguese (0.735), although on Student-Portuguese it was nearly tied with EW-KDSUM (0.734). The AUC advantage table is reported in Appendix C.

7 Discussion

The central message of this paper is that KDSUM can be used as a native PSD kernel for mixed-type embedding. The original KDSUM construction was developed as a similarity and dissimilarity for clustering. Here the same component kernels, once bandwidths are fixed, define a sample kernel that is PSD before centering and remains PSD after centering by $\boldsymbol{H}$. This moves the construction from a distance-based clustering tool to an admissible kernel geometry for PCO, KPCA, and fixed-$\boldsymbol{Q}$ PPCO. The distinction is important because a mixed-type dissimilarity can be useful for comparison while still producing an indefinite double-centered matrix. In that case the embedding requires truncation or spectral repair. KDSUM avoids this ambiguity by supplying the inner-product object directly.

The role of GA-KDSUM is to make this kernel geometry adaptive without losing PSD validity. Equal weighting treats all trace-balanced components as equally informative, which can be unstable when many variables are weakly related to the shared low-dimensional structure. The globally aligned weights use agreement with the consensus mixed-type geometry to increase the contribution of aligned components and reduce the influence of weakly aligned components. Because the final kernel is still a nonnegative combination of PSD component kernels, the weighting step changes the geometry while preserving the kernel guarantees. This explains why the clearest simulation gains appear in the setting with many irrelevant variables, where GA-KDSUM achieves the strongest latent factor recovery among the pairwise geometry baselines. It also explains why the method is not expected to dominate in every setting. When ordered variables directly encode the latent factors and few nuisance variables are present, ordinal Gower PCO can remain highly competitive. An unsupervised alignment rule can also emphasize a mutually aligned nuisance block, underweight isolated but useful variables, or miss task-specific supervised signal.

The Gower comparisons sharpen the interpretation of the results. Gower-type distances remain strong and useful baselines for heterogeneous data, but they do not necessarily produce PSD centered matrices. PSD-corrected Gower removes negative eigenvalues and makes PCO feasible, yet the simulations show that this repair does not by itself improve latent recovery. The same pattern appears in the downstream experiments, where GA-KDSUM improves over both standard Gower PCO and PSD-corrected Gower PCO. These results suggest that the advantage comes from constructing a type-aware PSD kernel and learning an aligned variable-weighted geometry, not merely from forcing a matrix to be PSD after the fact.

The real-data experiments support the same conclusion beyond synthetic latent factors. The Student Performance tasks contain continuous, unordered categorical, and ordered categorical variables, and the evaluation embeds test observations only through out-of-sample projection under repeated inductive cross-validation. On both tasks, GA-KDSUM gives the highest mean balanced accuracy and AUC. This shows that the proposed geometry can improve held-out predictive embeddings in real mixed-type data. The evidence is still limited to two related educational datasets, so broader benchmarks are needed before claiming broad empirical dominance across domains.

Several limitations remain. The current implementation forms dense pairwise matrices, so larger datasets require scalable kernel approximations, landmark or blockwise construction, or scalable fixed-$\boldsymbol{Q}$ PPCO solvers. The experiments also use fixed bandwidth choices and a fixed embedding dimension q, leaving systematic comparison of MSCV, task-tuned bandwidths, and data-driven choices of q for future work. Broader evaluations should compare GA-KDSUM with raw-feature factorization, supervised representation learning, and task-specific clustering or prediction methods, while also addressing missingness, high-cardinality categorical variables, and mixed local and global structure.

This discussion frames GA-KDSUM as a principled PSD mixed-type embedding method. Its key value is the combination of type-aware kernel construction, unsupervised geometry alignment, and positive semidefinite validity in one pairwise representation.

8 Conclusion

This paper establishes GA-KDSUM as a positive semidefinite kernel framework for mixed-type dimensionality reduction. Starting from the KDSUM similarity, we show that Gaussian, Aitken, and Wang–van Ryzin component kernels define valid PSD sample kernels after the bandwidths are

fixed, and that their nonnegative combinations preserve PSD structure. This result moves KDSUM beyond its original role as a mixed-type dissimilarity for clustering and turns it into a native kernel geometry for PCO, KPCA, and fixed-$\boldsymbol{Q}$ PPCO.

The globally aligned extension strengthens this kernel construction by making the geometry adaptive to shared structure. Through trace normalization and nonnegative alignment weights, GA-KDSUM combines type-aware treatment of continuous, unordered categorical, and ordered categorical variables with automatic downweighting of weakly aligned variables. It preserves positive semidefiniteness without post-hoc spectral correction. The method therefore does not merely repair a mixed-type distance for embedding. It constructs a valid and adaptive mixed-type kernel from the outset.

The empirical results support this interpretation. In simulations, GA-KDSUM is most effective when informative mixed-type variables are embedded among many irrelevant variables, precisely the regime where equal weighting and generic mixed-type distances are expected to struggle. In the Student Performance benchmarks, the same kernel embedding pipeline achieves the best held-out balanced accuracy and AUC among the compared pairwise geometry methods. These results indicate that the gains come from learning a more informative mixed-type geometry while maintaining PSD validity.

In general, GA-KDSUM provides a practical and principled route from heterogeneous tabular data to kernel-based low-dimensional representations. It avoids homogenizing variable types, avoids relying on indefinite dissimilarities, and avoids separating PSD correction from variable relevance weighting. This makes it a strong foundation for mixed-type spectral embedding, kernel PCA, and probabilistic principal coordinate analysis in modern tabular settings where heterogeneous variables, irrelevant features, and downstream embedding requirements appear together.

Acknowledgements

I would like to thank Daniel Krasnov for his mentorship, guidance, and helpful feedback throughout this project.

References

Aitchison, J. and Aitken, C. G. G. (1976). Multivariate binary discrimination by the kernel method. *Biometrika*, 63(3), 413–420.

Borg, I. and Groenen, P. J. F. (1997). *Modern Multidimensional Scaling: Theory and Applications*. Springer.

Cortes, C., Mohri, M., and Rostamizadeh, A. (2012). Algorithms for learning kernels based on centered alignment. *Journal of Machine Learning Research*, 13, 795–828.

Cortez, P. (2008). Student Performance [Dataset]. UCI Machine Learning Repository. https://doi.org/10.24432/C5TG7T.

Cortez, P. and Silva, A. M. G. (2008). Using data mining to predict secondary school student performance. In *Proceedings of the 5th Annual Future Business Technology Conference*, 5–12.

Eckart, C. and Young, G. (1936). The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3), 211–218.

Foss, A. H., Markatou, M., Ray, B., and Heching, A. (2016). A semiparametric method for clustering mixed data. *Machine Learning*, 105(3), 419–458.

Foss, A. H., Markatou, M., and Ray, B. (2019). Distance metrics and clustering methods for mixed-type data. *International Statistical Review*, 87(1), 80–109.

Ghashti, J. S. and Thompson, J. R. J. (2025). Mixed-type distance shrinkage and selection for clustering via kernel metric learning. *Journal of Classification*, 42(2), 311–334. https://doi.org/10.1007/s00357-024-09493-z.

Gower, J. C. (1966). Some distance properties of latent root and vector methods used in multivariate analysis. *Biometrika*, 53(3/4), 325–338. https://doi.org/10.1093/biomet/53.3-4.325.

Gower, J. C. (1971). A general coefficient of similarity and some of its properties. *Biometrics*, 27(4), 857–871.

Huang, Z. (1998). Extensions to the k-means algorithm for clustering large data sets with categorical values. *Data Mining and Knowledge Discovery*, 2(3), 283–304.

Horn, R. A. and Johnson, C. R. (2012). *Matrix Analysis*. Second edition. Cambridge University Press.

Jolliffe, I. T. (2002). *Principal Component Analysis*. Second edition. Springer.

McParland, D. and Gormley, I. C. (2016). Model based clustering for mixed data: clustMD. *Advances in Data Analysis and Classification*, 10(2), 155–169.

Miotto, R., Li, L., Kidd, B. A., and Dudley, J. T. (2016). Deep Patient: An unsupervised representation to predict the future of patients from the electronic health records. *Scientific Reports*, 6, 26094.

Podani, J. (1999). Extending Gower's general coefficient of similarity to ordinal characters. *Taxon*, 48(2), 331–340.

Schölkopf, B., Smola, A., and Müller, K.-R. (1998). Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10(5), 1299–1319.

Wang, M.-C. and Van Ryzin, J. (1981). A class of smooth estimators for discrete distributions. *Biometrika*, 68(1), 301–309.

Williams, C. K. I. (2002). On a connection between kernel PCA and metric multidimensional scaling. *Machine Learning*, 46, 11–19.

Yeh, I.-C. and Lien, C.-H. (2009). The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients. *Expert Systems with Applications*, 36(2), 2473–2480.

Zhang, Z. and Jordan, M. I. (2009). Latent variable models for dimensionality reduction. In *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics*, 655–662.

A Proofs

A.1 PSD details used in Proposition 1

This subsection records the two standard PSD facts used in Proposition 1.

First, the Gaussian component is PSD. Let

$$\boldsymbol{G}_{ij} = \exp\left\{-\frac{(x_i - x_j)^2}{2\lambda^2}\right\}, \qquad \lambda > 0.$$

For any $\boldsymbol{a} = (a_1, \ldots, a_n)^\top \in \mathbb{R}^n$,

$$\begin{aligned}\boldsymbol{a}^\top \boldsymbol{G}\boldsymbol{a} &= \sum_{i,j=1}^{n} a_i a_j \exp\left\{-\frac{x_i^2}{2\lambda^2}\right\} \exp\left\{-\frac{x_j^2}{2\lambda^2}\right\} \exp\left\{\frac{x_i x_j}{\lambda^2}\right\} \\ &= \sum_{m=0}^{\infty} \frac{1}{m!\lambda^{2m}} \left(\sum_{i=1}^{n} a_i \exp\left\{-\frac{x_i^2}{2\lambda^2}\right\} x_i^m\right)^2 \geq 0.\end{aligned}$$

Thus $\boldsymbol{G} \succeq 0$.

Since $(\lambda\sqrt{2\pi})^{-1} > 0$, the normalized Gaussian component matrix is also PSD.

Second, the matrix $\boldsymbol{T}_\lambda$ with entries $(\boldsymbol{T}_\lambda)_{rs} = \lambda^{|r-s|}$ is PSD for every $\lambda \in [0, 1]$. For $0 \leq \lambda < 1$, let $\varepsilon_1, \ldots, \varepsilon_G$ be independent standard normal random variables and define

$$Z_1 = \varepsilon_1, \qquad Z_t = \lambda Z_{t-1} + \sqrt{1 - \lambda^2}\, \varepsilon_t, \qquad t = 2, \ldots, G.$$

Then $\mathrm{Var}(Z_t) = 1$ and, for $r \leq s$,

$$\mathrm{Cov}(Z_r, Z_s) = \lambda^{s-r}.$$

Hence $\boldsymbol{T}_\lambda$ is a covariance matrix and is PSD. For $\lambda = 1$, $\boldsymbol{T}_\lambda = \mathbf{1}_G \mathbf{1}_G^\top$, which is PSD. □

A.2 Proof of Corollary 1

Proof. By Proposition 1, $\boldsymbol{\Psi}_{\mathrm{KDSUM}} \succeq 0$. Hence there exists a matrix $\boldsymbol{B}$ such that

$$\boldsymbol{\Psi}_{\mathrm{KDSUM}} = \boldsymbol{B}\boldsymbol{B}^\top.$$

Let $\boldsymbol{\phi}_i^\top$ be the ith row of $\boldsymbol{B}$. Then

$$(\boldsymbol{\Psi}_{\mathrm{KDSUM}})_{ij} = \langle \boldsymbol{\phi}_i, \boldsymbol{\phi}_j \rangle.$$

Therefore

$$\begin{aligned}\delta_{ij}^2 &= (\boldsymbol{\Psi}_{\mathrm{KDSUM}})_{ii} + (\boldsymbol{\Psi}_{\mathrm{KDSUM}})_{jj} - 2(\boldsymbol{\Psi}_{\mathrm{KDSUM}})_{ij} \\ &= \|\boldsymbol{\phi}_i\|^2 + \|\boldsymbol{\phi}_j\|^2 - 2\langle \boldsymbol{\phi}_i, \boldsymbol{\phi}_j \rangle \\ &= \|\boldsymbol{\phi}_i - \boldsymbol{\phi}_j\|^2.\end{aligned}$$

Thus δ_{ij}^2 is a squared Hilbert space distance.

Let $\boldsymbol{d} = \mathrm{diag}(\boldsymbol{\Psi}_{\mathrm{KDSUM}})$ be the vector of diagonal entries of $\boldsymbol{\Psi}_{\mathrm{KDSUM}}$. Then

$$\boldsymbol{\Delta} = \boldsymbol{d}\mathbf{1}_n^\top + \mathbf{1}_n \boldsymbol{d}^\top - 2\boldsymbol{\Psi}_{\mathrm{KDSUM}}.$$

Using $\boldsymbol{H}\mathbf{1}_n = 0$ and $\mathbf{1}_n^\top \boldsymbol{H} = 0^\top$, we obtain

$$\begin{aligned} -\frac{1}{2}\boldsymbol{H}\boldsymbol{\Delta}\boldsymbol{H} &= -\frac{1}{2}\boldsymbol{H}\left(\boldsymbol{d}\mathbf{1}_n^\top + \mathbf{1}_n\boldsymbol{d}^\top - 2\boldsymbol{\Psi}_{\mathrm{KDSUM}}\right)\boldsymbol{H} \\ &= \boldsymbol{H}\boldsymbol{\Psi}_{\mathrm{KDSUM}}\boldsymbol{H}. \end{aligned}$$

This proves the claim. □

B Additional Simulation Results

This appendix contains secondary simulation diagnostics and illustrative figures omitted from the main text. The main simulation is the mixed-type shared-signal design described in Section 5. All final simulation tables use $B = 100$, $n = 180$, and embedding dimension $q = 2$.

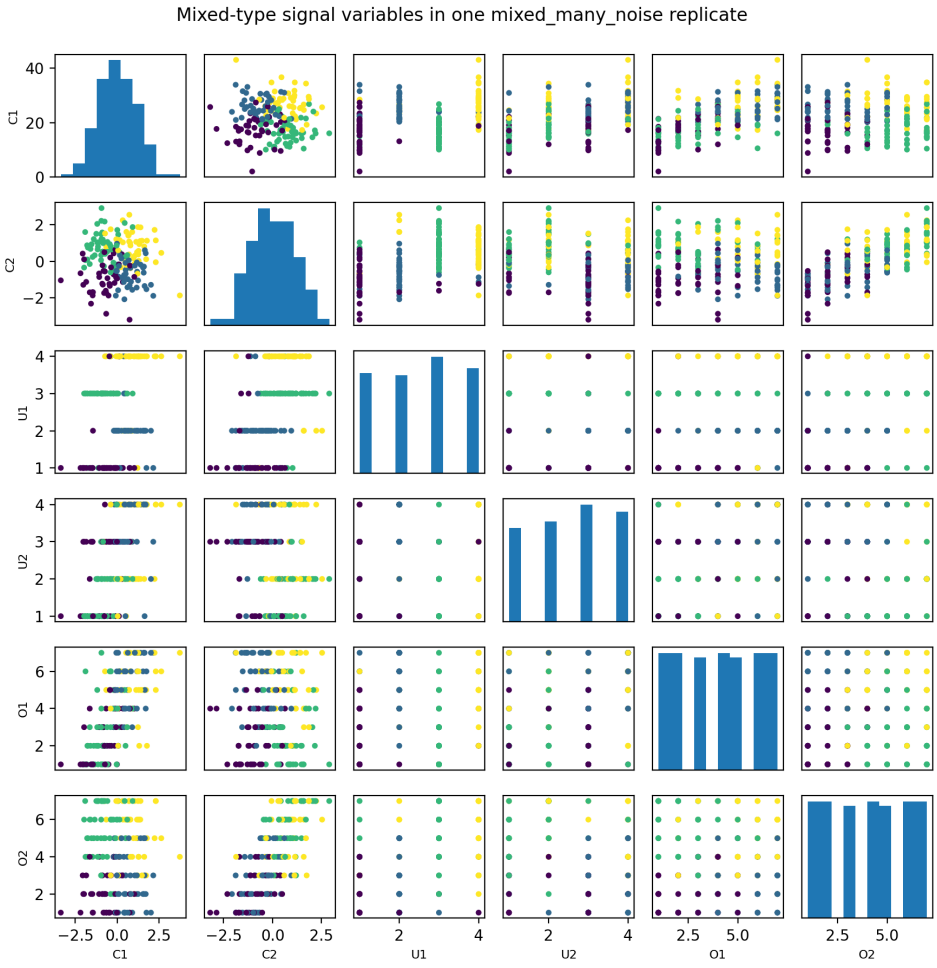

Figure 3: Pair plot of representative observed variables in one mixed-signal replicate. Points are colored by latent quadrant only for visualization.

Table 7: Full latent factor recovery across simulation scenarios. Larger R^2_{latent} is better.

Scenario	Method	Mean R^2_{latent}	SE
mixed clean	GA-KDSUM	0.758	0.004
mixed clean	EW-KDSUM	0.831	0.002
mixed clean	Gower PCO	0.830	0.002
mixed clean	ordinal Gower PCO	0.868	0.001
mixed clean	PSD-corrected Gower PCO	0.830	0.002
mixed noise	GA-KDSUM	0.794	0.002
mixed noise	EW-KDSUM	0.823	0.002
mixed noise	Gower PCO	0.811	0.002
mixed noise	ordinal Gower PCO	0.844	0.002
mixed noise	PSD-corrected Gower PCO	0.811	0.002
mixed high noise	GA-KDSUM	0.796	0.002
mixed high noise	EW-KDSUM	0.793	0.002
mixed high noise	Gower PCO	0.767	0.002
mixed high noise	ordinal Gower PCO	0.803	0.002
mixed high noise	PSD-corrected Gower PCO	0.767	0.002
mixed many noise	GA-KDSUM	0.821	0.002
mixed many noise	EW-KDSUM	0.769	0.002
mixed many noise	Gower PCO	0.736	0.003
mixed many noise	ordinal Gower PCO	0.781	0.002
mixed many noise	PSD-corrected Gower PCO	0.736	0.003

Table 8: Best method by simulation scenario. This table highlights that GA-KDSUM is strongest in the many-noise setting but not uniformly best in clean or mild-noise settings.

Scenario	Best method	Best R^2_{latent}	GA-KDSUM R^2_{latent}
mixed clean	ordinal Gower PCO	0.868	0.758
mixed noise	ordinal Gower PCO	0.844	0.794
mixed high noise	ordinal Gower PCO	0.803	0.796
mixed many noise	GA-KDSUM	0.821	0.821

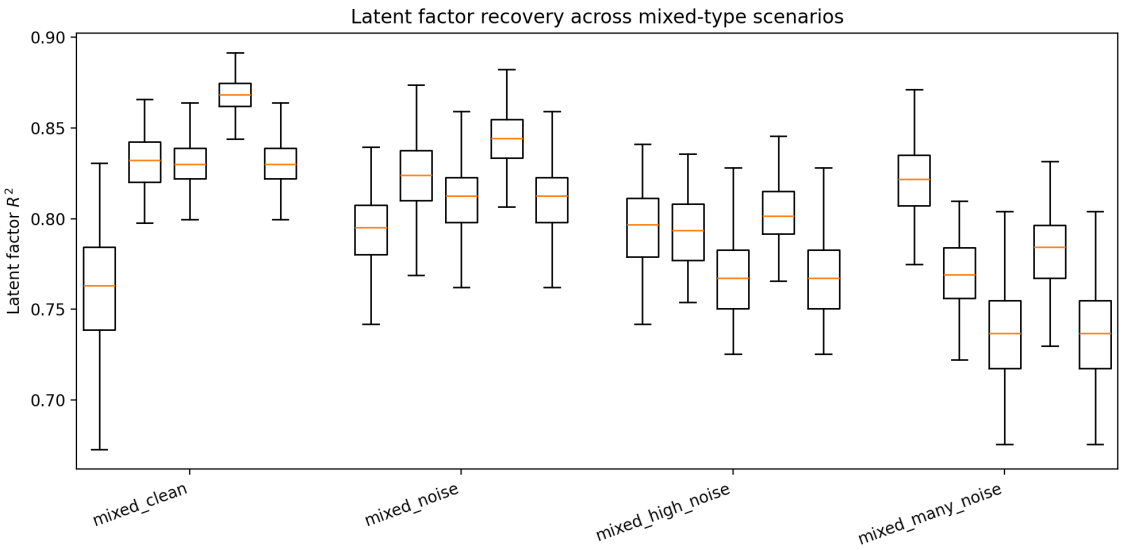

Figure 4: Latent factor R^2 boxplots by scenario and method.

Table 9: Full paired latent factor R^2 difference relative to Gower PCO. Positive values favor the listed method.

Scenario	Method	ΔR^2 vs. Gower	SE	Win rate
mixed clean	GA-KDSUM	-0.072	0.004	0.00
mixed clean	EW-KDSUM	0.001	0.001	0.59
mixed clean	ordinal Gower PCO	0.038	0.000	1.00
mixed clean	PSD-corrected Gower PCO	0.000	0.000	0.16
mixed noise	GA-KDSUM	-0.017	0.002	0.23
mixed noise	EW-KDSUM	0.013	0.001	0.94
mixed noise	ordinal Gower PCO	0.033	0.001	1.00
mixed noise	PSD-corrected Gower PCO	0.000	0.000	0.18
mixed high noise	GA-KDSUM	0.028	0.002	0.92
mixed high noise	EW-KDSUM	0.026	0.001	0.98
mixed high noise	ordinal Gower PCO	0.035	0.001	1.00
mixed high noise	PSD-corrected Gower PCO	0.000	0.000	0.14
mixed many noise	GA-KDSUM	0.085	0.002	1.00
mixed many noise	EW-KDSUM	0.034	0.002	0.99
mixed many noise	ordinal Gower PCO	0.046	0.001	1.00
mixed many noise	PSD-corrected Gower PCO	0.000	0.000	0.20

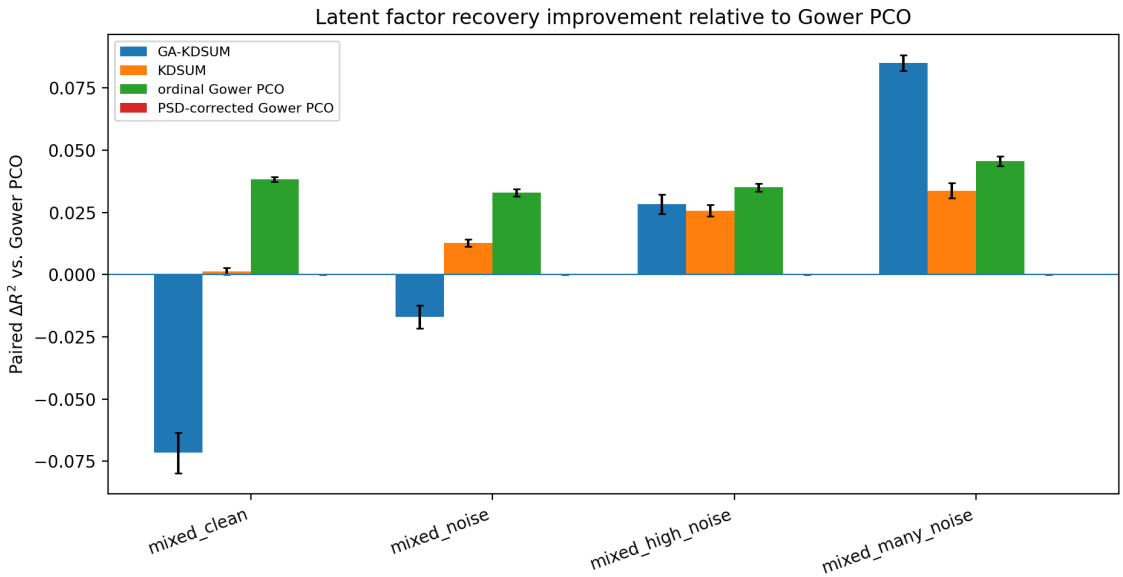

Figure 5: Paired ΔR^2 relative to Gower PCO across all simulation scenarios. Error bars show ± 1.96 Monte Carlo standard errors.

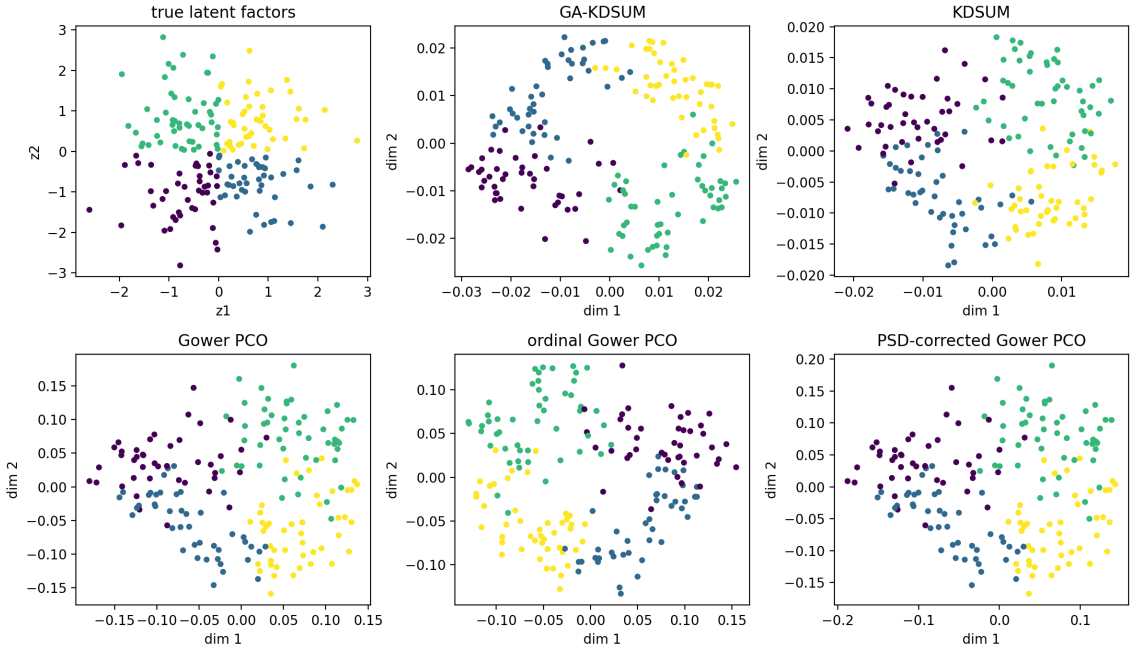

Figure 6: Embedding comparison for one mixed-signal replicate. Colors indicate latent quadrants and are used only for visualization.

Table 10: Mean total kernel weight by variable type and by signal/noise status. In this simulation, signal components include continuous, unordered, and ordered variables generated from the latent factors.

Scenario	Method	Continuous	Unordered	Ordered	Signal weight	Noise weight
mixed clean	GA-KDSUM	0.219	0.664	0.117	1.000	0.000
mixed clean	EW-KDSUM	0.286	0.429	0.286	1.000	0.000
mixed noise	GA-KDSUM	0.266	0.535	0.199	0.805	0.195
mixed noise	EW-KDSUM	0.316	0.368	0.316	0.368	0.632
mixed high noise	GA-KDSUM	0.243	0.464	0.293	0.615	0.385
mixed high noise	EW-KDSUM	0.323	0.355	0.323	0.226	0.774
mixed many noise	GA-KDSUM	0.224	0.391	0.385	0.256	0.744
mixed many noise	EW-KDSUM	0.328	0.343	0.328	0.104	0.896

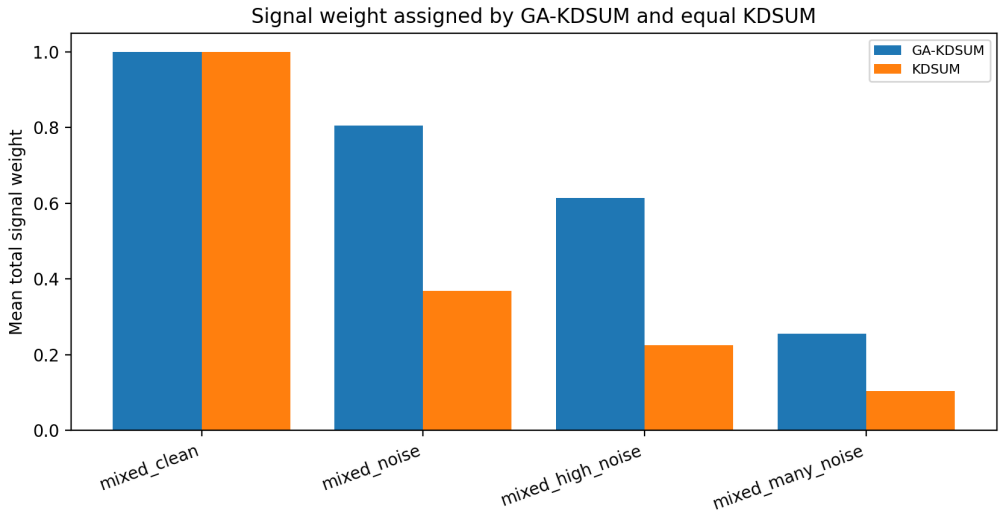

Figure 7: Mean signal weight assigned by GA-KDSUM and the equal-weight trace-balanced baseline EW-KDSUM across simulation scenarios.

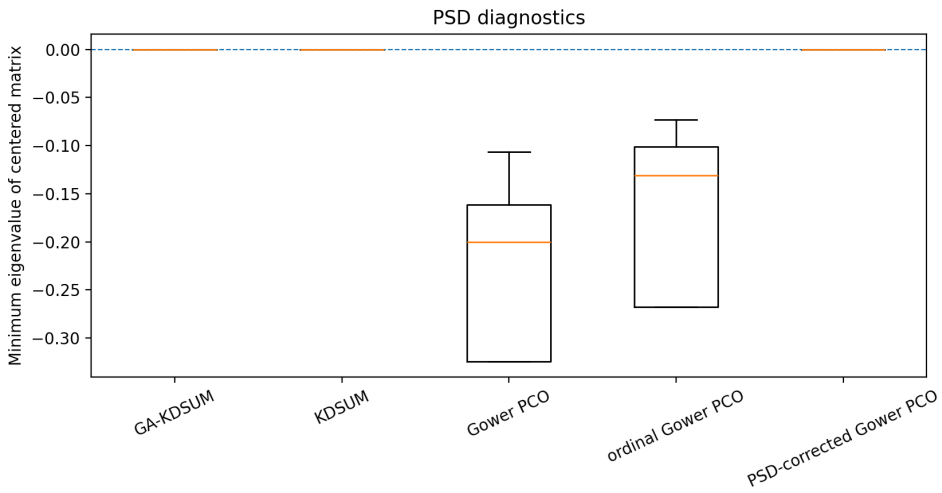

Figure 8: Minimum eigenvalue of the centered KDSUM kernel matrix $\boldsymbol{Q}$ for the equal-weight trace-balanced baseline EW-KDSUM and GA-KDSUM. Values are at numerical precision.

C Additional Real-Data Results

This appendix reports supplementary real-data metrics for the two Student Performance tasks used in the main text. The appendix is limited to results needed to support the discussion of the Student benchmark and does not include unrelated off-target datasets.

Table 11: Full real-data downstream performance under repeated five-fold inductive cross-validation. The primary score is balanced accuracy.

Dataset	Method	Bal. acc.	AUC	Rank
Student-Math	GA-KDSUM	0.661 (0.012)	0.699 (0.013)	1
Student-Math	EW-KDSUM	0.605 (0.011)	0.657 (0.013)	2
Student-Math	ordinal Gower PCO	0.578 (0.013)	0.607 (0.014)	3
Student-Math	PSD-corrected Gower PCO	0.555 (0.014)	0.600 (0.014)	4
Student-Math	Gower PCO	0.553 (0.014)	0.600 (0.014)	5
Student-Portuguese	GA-KDSUM	0.703 (0.014)	0.735 (0.017)	1
Student-Portuguese	EW-KDSUM	0.677 (0.015)	0.734 (0.018)	2
Student-Portuguese	PSD-corrected Gower PCO	0.645 (0.015)	0.678 (0.019)	3
Student-Portuguese	Gower PCO	0.639 (0.015)	0.679 (0.019)	4
Student-Portuguese	ordinal Gower PCO	0.621 (0.016)	0.657 (0.017)	5

Table 12: Paired balanced-accuracy difference relative to Gower PCO on the same cross-validation splits. Positive values favor the listed method.

Dataset	Method	Δ bal. acc. vs. Gower	SE	Win rate
Student-Math	GA-KDSUM	0.108	0.010	0.92
Student-Math	EW-KDSUM	0.052	0.008	0.84
Student-Math	ordinal Gower PCO	0.025	0.007	0.76
Student-Math	PSD-corrected Gower PCO	0.001	0.001	0.12
Student-Portuguese	GA-KDSUM	0.064	0.015	0.84
Student-Portuguese	EW-KDSUM	0.038	0.014	0.68
Student-Portuguese	ordinal Gower PCO	-0.019	0.012	0.32
Student-Portuguese	PSD-corrected Gower PCO	0.006	0.001	0.48

Table 13: Paired AUC advantage of GA-KDSUM over each real-data baseline. Positive values favor GA-KDSUM.

Dataset	Baseline	GA-KDSUM AUC − baseline	SE	Win rate
Student-Math	EW-KDSUM	0.043	0.007	0.96
Student-Math	Gower PCO	0.099	0.009	0.96
Student-Math	ordinal Gower PCO	0.092	0.012	0.92
Student-Math	PSD-corrected Gower PCO	0.099	0.009	0.96
Student-Portuguese	EW-KDSUM	0.002	0.008	0.44
Student-Portuguese	Gower PCO	0.056	0.011	0.92
Student-Portuguese	ordinal Gower PCO	0.078	0.012	0.92
Student-Portuguese	PSD-corrected Gower PCO	0.057	0.012	0.92

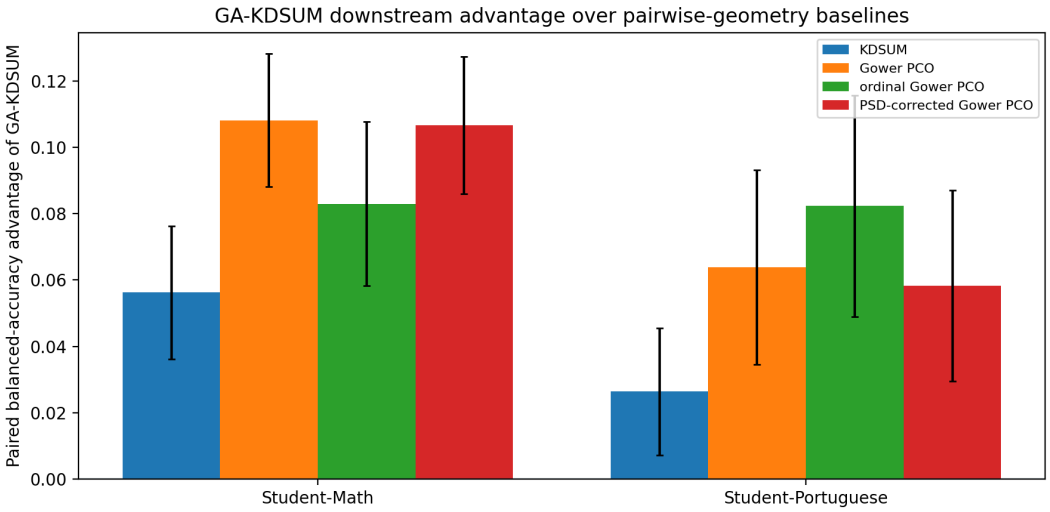

Figure 9: Paired balanced-accuracy advantage of GA-KDSUM over real-data baselines. Error bars show ±1.96 standard errors across repeated cross-validation splits.

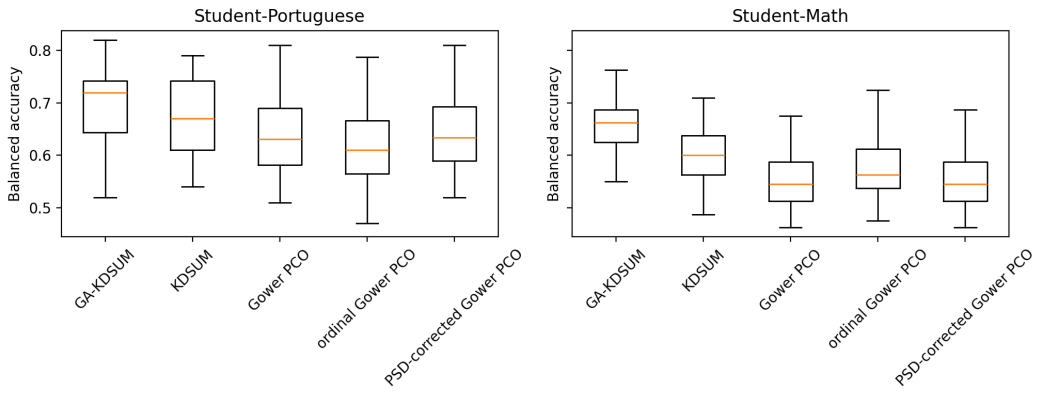

Figure 10: Balanced-accuracy distributions across repeated cross-validation splits for the two Student Performance tasks.