The Structured Latent Basis: Feature Engineering as Basis Selection

A Diagnostic Framework from Tabular to Text

Tamás Nagy, Ph.D.

ORCID: 0009-0004-8079-4679

Independent Researcher

tnagyphd@gmail.com

https://thalens.org

Working Paper • August 2026

Build-std v2.0+1af679ec50 | 2026-08-08 20:32 | dfcb5a15

Overview

Many supervised learning pipelines share a representation stage and a prediction stage. There is an input space $\mathcal{X}$, a target y, and a function $f : \mathcal{X} \to \mathbb{R}$ to estimate. A conventional pipeline calls the first stage **feature engineering** (transform $\mathcal{X}$ into a feature space $\mathcal{F}$) and the second **modeling** (fit a predictor in $\mathcal{F}$). These stages are often treated as separate crafts—domain heuristics for the features, followed by gradient boosting, kernel methods, or neural networks for prediction.

This paper studies one useful joint formulation: **choose a feature dictionary in which the target can be approximated by a regularized linear predictor**.

If the dictionary is well chosen, the target may admit an approximation $y \approx \sum_i w_i b_i(\mathbf{x})$, where $\{b_i\}$ are structured features and $\mathbf{w}$ is estimated by Ridge regression. This relocates part—not necessarily all—of the learning difficulty into representation design. In this view, feature construction and predictor fitting must be assessed together.

We call this the **Structured Latent Basis (SLB)** framework. The feature system is *structured* when its construction reflects a hypothesized property of the target (smooth, threshold, periodic, or sequential); it is *latent* because the useful representation is not directly observed and must be proposed or learned from data and domain knowledge.

This is not only a metaphor. We develop two precise mathematical statements and three application paths:

- **Conditional Ridge optimality:** Once a finite feature matrix and positive regularization level are fixed, the Ridge objective has a unique global minimizer in closed form.
- **Structured approximation:** For targets consisting of an analytic cosine-decay component plus finitely many jumps, a truncated cosine dictionary plus logistic step features has an explicit approximation bound.
- **Applications:** S³M supplies a tabular construction; DCT features supply a text hypothesis; and linear-probe gaps provide an empirical diagnostic. The text and probe experiments are currently preregistered for leakage-safe replication rather than presented as verified findings.

The framework proposes a falsifiable research program: for a specified task family, search for a structured feature system in which a regularized linear predictor closes most of the gap to stronger nonlinear baselines. The basis encodes part of what the model would otherwise have to learn. Neural networks learn implicit representations through backpropagation; the SLB approach asks which useful parts can be constructed explicitly and audited.

Abstract

We introduce the Structured Latent Basis (SLB) framework, a perspective that compares feature engineering, meta-learning, and representation learning through a common diagnostic question: does the chosen representation make a regularized linear predictor sufficient under a declared evaluation protocol?

We make two formal observations and define an empirical program:

1. **Conditional optimality:** For a fixed feature matrix and $\alpha > 0$, the regularized linear objective is strictly convex and Ridge gives its unique minimizer.
2. **Approximation:** An analytic cosine-decay component plus finitely many jumps admits a quantitative cosine-plus-sigmoid approximation bound.
3. **Diagnostic protocol:** Compare a regularized linear probe with specified nonlinear baselines after all learned preprocessing is fit inside the training fold. The current text implementation establishes the fold-local Ridge path and valid-token feature extraction; nonlinear baselines remain part of the prospective protocol, and the required embedding caches must be regenerated before numerical claims are reinstated.

The unifying insight is narrow: feature construction and model fitting should be evaluated jointly. The basis can be hand-designed, inherited from pre-training, or learned across tasks. A companion paper studies learned feature maps with closed-form Ridge adaptation and explicitly relates that mechanism to R2-D2. The present paper does not claim that a suitable basis always exists at practical dimension or that Ridge is globally optimal among model classes.

Keywords: feature engineering, basis selection, meta-learning, spectral methods, Ridge regression, few-shot learning, representation learning

---

1. Introduction

1.1 The Two-Stage View

Supervised learning pipelines have a standard structure:

$$\mathbf{x} \xrightarrow{\text{feature engineering}} \mathbf{z} \xrightarrow{\text{model}} \hat{y}$$

The first arrow is "feature engineering" — domain-specific transformations, interaction terms, log transforms, one-hot encoding, embedding lookups. The second arrow is "the model" — XGBoost, neural network, random forest, logistic regression.

These two stages are conventionally treated as independent. Feature engineering is a craft; modeling is an optimization problem. Tabular pipelines often pair feature engineering with gradient boosting, including XGBoost (Chen & Guestrin, 2016). NLP breakthroughs came from learning embeddings that serve as features for downstream classifiers.

The separation is operationally useful but statistically coupled. A representation determines which predictors are simple or data-efficient, while the downstream model determines which properties of that representation can be used. Some feature constructions make a target approximately linear in the transformed space; in those cases, a regularized linear model may suffice.

1.2 The Basis Selection Perspective

We formalize this observation. Given training data $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$, the learning problem is to approximate $f(\mathbf{x}) = \mathbb{E}[y \mid \mathbf{x}]$. Consider a collection of basis functions $\{b_1, b_2, \ldots, b_N\}$ where each $b_k : \mathcal{X} \to \mathbb{R}$. The model is:

$$\hat{f}(\mathbf{x}) = \sum_{k=1}^{N} w_k \cdot b_k(\mathbf{x})$$

with weights $\mathbf{w}$ found by Ridge regression (Hastie et al., 2009):

$$\mathbf{w}^* = \arg\min_{\mathbf{w}} \sum_{i=1}^{n} (y_i - \mathbf{B}_i \mathbf{w})^2 + \alpha \|\mathbf{w}\|^2$$

The quality of the approximation depends primarily on the chosen feature family, but also on finite-sample estimation factors (sample size, noise, conditioning, and regularization). If $\{b_k\}$ spans the relevant function class and the estimation problem is well-conditioned at the available n, the Ridge solution will be accurate. If not, no amount of regularization can fully compensate.

SLB working hypothesis. Feature engineering can often be treated as feature-dictionary selection. In orthogonal settings, a useful dictionary may yield sparse or rapidly decaying coefficients; in redundant or finite-sample settings, stable approximation and estimation are the relevant criteria.

This is the Structured Latent Basis working principle. A feature family is *structured* when it encodes a stated hypothesis about smoothness, periodicity, threshold structure, or sequential organization. It is *latent* because its usefulness is not directly observed and must be evaluated from finite data.

1.3 Why "Latent"?

Neural networks learn feature maps through their hidden layers. When a trained network has the form $f(\mathbf{x}) = \mathbf{w}^\top \phi_\theta(\mathbf{x})$, it decomposes prediction into a learned representation ϕ_θ and a linear head $\mathbf{w}$. Calling the coordinates of ϕ_θ a "basis" is a useful analogy, not a claim that they form an independent or complete mathematical basis.

The SLB framework asks: can we construct the basis *explicitly*, using mathematical knowledge about the function class, rather than learning it implicitly through gradient descent?

In several domains, explicit dictionaries provide plausible candidates:

- For **smooth functions with a compatible even extension**: cosine modes $\cos(k\pi x)$ can have rapidly, and under suitable analyticity assumptions geometrically, decaying coefficients (Trefethen, 2013).
- For **threshold functions**: sigmoid features $\sigma((x-t)/\varepsilon)$ provide a smooth bridge that converges pointwise away from the threshold to a step as $\varepsilon \to 0$; the companion paper *The Smooth-Step Spectral Method: Unifying Smooth and Threshold Structure in Tabular Regression* develops this construction (Nagy, 2026).
- For **text sequences**: the Discrete Cosine Transform on embedding sequences decomposes the signal into positional frequency components; any association between bands and linguistic semantics is treated as an empirical hypothesis tested in §4.

Each motivates the same testable principle: *match the feature dictionary to the function class, then measure whether a regularized linear head is sufficient.*

1.4 This Paper's Contribution

1. We formalize the SLB framework as a unifying perspective on feature engineering across modalities (§2).
2. We prove conditional Ridge optimality and a restricted cosine-plus-sigmoid approximation bound (§§2–3).
3. We specify a leakage-safe text benchmark and withdraw earlier exploratory numbers that do not satisfy that protocol (§4).
4. We position learned-basis adaptation as a companion direction rather than a new mechanism: differentiable Ridge meta-learning is established prior art in R2-D2 (§6).
5. We define the nonlinear-model gap as a diagnostic of linear-probe sufficiency on pre-trained embeddings (§7).
6. We synthesize these components into a bounded research program rather than a universal claim (§8).

1.5 Non-Claims

- We do not claim SLB replaces deep learning. For complex vision, audio, or reinforcement learning tasks, learned representations remain essential. SLB is most powerful when the function class has known mathematical structure.
- The text instantiation is a preregistered proof-of-concept protocol. Earlier exploratory numbers are excluded from the paper's evidential core until every learned transform is fit inside the outer training fold and the caches are regenerated.
- We do not claim optimality of any specific basis. The framework proposes candidate feature families; the number of modes, bandwidth schedules, DCT truncation, and related choices remain hyperparameters.

1.6 Relationship to Prior Work

The individual mechanisms used here have substantial prior art. Supervised dictionary learning already couples a learned shared dictionary or sparse representation to a downstream discriminative predictor (Mairal et al., 2008; Gangeh et al., 2015), and basis selection over prescribed dictionaries is a classical sparse-approximation problem. DCT sentence representations were introduced by Almarwani et al. (2019), and later work applied learnable spectral filters to contextual embeddings (Müller-Eberstein et al., 2022). FNet uses Fourier mixing inside a Transformer architecture (Lee-

Thorp et al., 2022), which is related motivation but not evidence for any particular smoothness law of token embeddings. In meta-learning, R2-D2 already trains a feature extractor through a differentiable closed-form Ridge solver (Bertinetto et al., 2019), and subsequent work analyzes why meta-learned representations support simple last-layer adaptation (Goldblum et al., 2020).

Accordingly, this paper does not claim invention of feature-dictionary learning or selection, DCT sentence features, linear probing, or differentiable Ridge adaptation. Its bounded contribution is a cross-modal, protocol-centered synthesis of explicit and inherited structured representations, together with a reproducible diagnostic: after fixing a representation, measure how much carefully tuned nonlinear models improve over a regularized linear probe.

---

2. The Structured Latent Basis Framework

2.1 Formal Setup

Let $(\mathcal{X}, \mu)$ be the input domain with measure μ, and let $f \in L^2(\mathcal{X}, \mu)$ be the target function. A **structured feature system** is a sequence $\mathcal{B} = \{b_k\}_{k \geq 1}$ of functions $b_k : \mathcal{X} \to \mathbb{R}$, used through finite truncations $\mathcal{B}_N = \{b_k\}_{k=1}^N$, with the following intended properties:

1. **Completeness (when applicable):** When the chosen family is complete for the relevant function class, its span can approximate f to any desired accuracy as $N \to \infty$.
2. **Rapid decay (idealized coefficient picture):** In regimes where an orthogonal-basis viewpoint is appropriate, the coefficients $w_k = \langle f, b_k \rangle$ decay rapidly (exponentially for analytic structure, polynomially for limited regularity). In the redundant/non-orthogonal dictionary setting, the practical criterion is instead stability and the existence of a well-conditioned finite-sample estimator.
3. **Interpretability (optional):** Individual b_k may have an intended meaning tied to the data domain, such as frequency, threshold location, or position scale.

The SLB model is:

$$\hat{f}_N(\mathbf{x}) = \sum_{k=1}^{N} w_k \cdot b_k(\mathbf{x}), \quad \mathbf{w} = \text{Ridge}(\mathbf{B}, \mathbf{y}, \alpha)$$

where $\mathbf{B} \in \mathbb{R}^{n \times N}$ is the basis matrix with $B_{ik} = b_k(\mathbf{x}_i)$.

Proposition 1 (conditional Ridge optimality). Fix a feature matrix $\mathbf{B}$, targets $\mathbf{y}$, and $\alpha > 0$. Then

$$L(\mathbf{w}) = \|\mathbf{y} - \mathbf{B}\mathbf{w}\|_2^2 + \alpha \|\mathbf{w}\|_2^2$$

has the unique minimizer

$$\mathbf{w}^* = (\mathbf{B}^\top \mathbf{B} + \alpha \mathbf{I})^{-1} \mathbf{B}^\top \mathbf{y}.$$

Proof. The Hessian is $2(\mathbf{B}^\top\mathbf{B} + \alpha\mathbf{I})$. For every nonzero v, $v^\top(\mathbf{B}^\top\mathbf{B} + \alpha\mathbf{I})v = \|\mathbf{B}v\|_2^2 + \alpha\|v\|_2^2 > 0$. Thus L is strictly convex. Setting its gradient to zero gives the displayed normal equation and its unique solution. □

This proposition is deliberately conditional. It says that Ridge is optimal for the stated regularized linear objective after the representation is fixed. It does not say that Ridge is the best predictor among all model classes or that a chosen representation is adequate.

2.2 The Basis Selection Principle

Definition. A basis $\mathcal{B}$ is *natural* for a function class $\mathcal{F}$ when functions in $\mathcal{F}$ admit controlled coefficient decay in $\mathcal{B}$. In an orthonormal setting, one useful pointwise-in-function formulation is:

$$\forall f \in \mathcal{F},\ \exists\, C_f > 0,\ \rho_f \in (0,1) \text{ such that } |w_k(f)| \le C_f \rho_f^k.$$

For a fixed f, this condition bounds the coefficient tail geometrically; the resulting truncation rate depends on the norm, basis normalization, and constants C_f, ρ_f. A uniform class-level rate requires uniform constants and additional assumptions. Redundant feature dictionaries need a different criterion based on stable finite-dimensional approximation rather than unique orthogonal coefficients.

Observation. Many successful feature-engineering strategies can be interpreted as constructing a useful feature dictionary:

Feature engineering technique	Implicit basis	Function class
Log-transform of price	$\{\log(x)\}$	Multiplicative relationships
Polynomial features	$\{x^k\}$	Polynomial targets
One-hot encoding	$\{\mathbf{1}_{x=c}\}$	Categorical effects
Fourier features	$\{e^{i\omega x}\}$	Periodic/smooth targets
Interaction terms	$\{x_i \cdot x_j\}$	Bilinear effects
TF-IDF	$\{w_t \cdot \text{idf}(t)\}$	Bag-of-words distributions

The SLB perspective does not invent these features. It supplies a hypothesis for why they work: they may provide coordinates in which the target is simpler to approximate and estimate.

2.3 When Does SLB Outperform Learned Representations?

The SLB approach has advantages when:

- **The function class is known.** Tabular data with smooth and threshold patterns → cosine + sigmoid basis. Sequential data with multi-scale structure → DCT basis.
- **Sample size is limited.** An a priori dictionary requires no training data to define; only its coefficients need estimation. Data-dependent thresholds, projections, and feature selection do require training data and must be learned inside the training fold.
- **Inspectability is required.** Explicit dictionaries can attach each coefficient to a named feature, although this does not by itself make the fitted model causally interpretable.

It has disadvantages when:

- **The function class is unknown or very complex.** Vision, audio, game-playing — the natural basis is not clear a priori.
- **The data is abundant.** Learned representations can be competitive when enough data and computation are available, although their success is task- and optimization-dependent.

2.4 Relationship to Kernel Methods

SLB is related to but distinct from kernel methods. A kernel $K(\mathbf{x}, \mathbf{x}')$ implicitly defines a feature map $\phi : \mathcal{X} \to \mathcal{H}$ into a Reproducing Kernel Hilbert Space. The SLB basis $\{b_k\}$ is an *explicit* finite-dimensional feature map. The distinction matters:

- Kernel methods have infinite-dimensional implicit features but finite-sample dual representation. SLB has finite explicit features.
- Standard exact kernel-matrix solvers typically use $O(n^2)$ memory and up to $O(n^3)$ factorization time. An explicit N-feature Ridge implementation uses $O(nN)$ design-matrix storage and, when solving the primal normal equations directly, roughly $O(nN^2 + N^3)$ time. The practical advantage therefore depends on $N \ll n$, numerical method, and conditioning.
- Explicit SLB features can be individually inspectable. Kernel representations may also be interpretable when their kernel or approximation has a clear structure.

Random Fourier features (Rahimi & Recht, 2007) bridge the two: they approximate a kernel with explicit random basis functions. SLB differs in that the basis functions are *structured* (designed from domain knowledge), not random.

---

3. Instantiation I: Tabular Data (S³M)

The tabular instantiation is developed fully in the companion paper *The Smooth-Step Spectral Method: Unifying Smooth and Threshold Structure in Tabular Regression* (Nagy, 2026). We summarize the key elements.

3.1 The Basis

For tabular data with features $\mathbf{x} = (x_1, \ldots, x_d) \in [0, 1]^d$, the S³M basis combines:

- **Cosine modes:** $\phi_{j,k}(\mathbf{x}) = \cos(k\pi x_j)$, $k = 1, \ldots, K$ per feature. Natural basis for smooth partial-dependence functions.
- **Multi-scale sigmoids:** $\psi_{j,m,s}(\mathbf{x}) = \sigma((x_j - t_{j,m})/(\varepsilon_s \cdot R_j))$ at data-learned thresholds $t_{j,m}$ and bandwidth scales $\varepsilon_s \in \{0.003, 0.01, 0.03, 0.1\}$. Natural basis for threshold effects.
- **Interaction products:** $\psi_i \cdot \psi_j$, $\psi_i \cdot \phi_j$, $\phi_i \cdot \psi_j$ for top correlated pairs.

3.2 A Restricted Approximation Bound

The following statement isolates the class for which the cosine-plus-sigmoid argument is valid. It is narrower than a claim about arbitrary piecewise-analytic functions.

Proposition 2 (analytic residual plus jumps). Let

$$f(x) = g(x) + \sum_{j=1}^{J} a_j H(x - t_j), \qquad x \in [0,1],$$

where H is the Heaviside step function. Suppose $g(x) = \sum_{k\geq 0} c_k \cos(k\pi x)$ with $|c_k| \leq Cq^k$ for some $C > 0$ and geometric-decay factor $q \in (0,1)$. Define $s_\varepsilon(u) = (1 + e^{-u/\varepsilon})^{-1}$ and

$$f_{K,\varepsilon}(x) = \sum_{k=0}^{K} c_k \cos(k\pi x) + \sum_{j=1}^{J} a_j s_\varepsilon(x - t_j).$$

Then

$$\|f - f_{K,\varepsilon}\|_{L^2([0,1])} \leq \frac{Cq^{K+1}}{1-q} + \sqrt{(2\log 2 - 1)\varepsilon} \sum_{j=1}^{J} |a_j|.$$

Proof. Since $|\cos(k\pi x)| \leq 1$, the cosine tail is bounded in $L^2([0,1])$ by its L^∞ bound, $\sum_{k>K} |c_k| \leq Cq^{K+1}/(1-q)$. For one step, extending the integration domain gives

$$\|H - s_\varepsilon\|^2_{L^2([0,1])} \leq 2\varepsilon \int_0^\infty \frac{dv}{(1+e^v)^2} = (2\log 2 - 1)\varepsilon.$$

Apply the triangle inequality to the weighted jump terms and combine the two bounds. □

The proposition requires the jump-subtracted residual g to have geometric cosine-coefficient decay. Generic piecewise-analytic functions need not satisfy this condition because derivative mismatches can leave algebraic tails. The result is an approximation statement; it does not establish statistical risk, conditioning, or superiority over tree-based estimators.

3.3 Empirical Status

The companion S³M draft contains exploratory comparisons against fixed tree baselines, together with ablations and sensitivity sweeps. Its current review status is HOLD: fold-level scores are not yet present, some comparisons are not nested, and the reported dataset count is inconsistent across drafts. Those tables are therefore not imported as evidence here.

The admissible conclusion of this section is mathematical and architectural. Proposition 2 explains why cosine and sigmoid features are complementary for the restricted analytic-residual-plus-jumps class. Whether the resulting finite dictionary improves prediction over tuned tree, kernel, or neural baselines remains an empirical question requiring a separate complete result bundle.

3.4 Required Tabular Replication

A publication-grade replication must freeze preprocessing, split construction, feature-selection rules, and tuning budgets before evaluation. It must report fold-level scores for S³M-only, the residual architecture, and all baselines; use nested selection whenever hyperparameters or thresholds are learned; and distinguish synthetic mechanism tests from real-data generalization. Until that artifact exists, this paper makes no tabular performance or significance claim.

4. Instantiation II: Text Data (Spectral Text)

4.1 The Key Insight: Text as a Wave

A text is a sequence of tokens. A pretrained transformer maps each token to an embedding vector: the text becomes a matrix $\mathbf{E} \in \mathbb{R}^{L \times d}$ where L is the sequence length and d is the embedding dimension.

This matrix is a discrete signal — a d-dimensional function sampled at L positions. Just as a time-domain audio signal can be decomposed into frequency components, the embedding sequence can be decomposed into positional frequency components via the Discrete Cosine Transform.

DCT-based sentence representations predate this work: Almarwani et al. (2019) used low-order coefficients as order-sensitive sentence embeddings, and Müller-Eberstein et al. (2022) learned task-specific spectral filters over contextual embeddings. The present protocol extends this line by specifying a broader feature comparison and a diagnostic gap between regularized linear and selected nonlinear downstream models; no numerical improvement is claimed in this version.

The DCT-II applied along the position axis:

$$\hat{E}_{k,j} = \alpha_k \sum_{\ell=0}^{L-1} E_{\ell,j} \cos\left(\frac{\pi(2\ell+1)k}{2L}\right), \quad k = 0, 1, \ldots, K-1,$$

where $\alpha_0 = L^{-1/2}$ and $\alpha_k = (2/L)^{1/2}$ for $k \geq 1$.

produces coefficients $\hat{E}_{k,j}$ for frequency index k and embedding dimension j. Note that in the standard orthonormal DCT-II convention the $k = 0$ mode uses a different normalization constant than $k > 0$; our implementation follows this convention. Accordingly, we treat $k = 0$ as the DC/constant component (proportional to the mean embedding up to normalization), rather than identifying it with the mean itself.

We interpret these positional frequency bands as candidate descriptors; any linguistic or semantic association is an empirical hypothesis for the downstream benchmark:

- $k = 0$ **(DC/constant component):** Overall direction of the embedding sequence (proportional to the position-wise mean up to normalization). Its incremental predictive value must be measured against the sentence-embedding and sequence-length controls.
- $k = 1$–5 **(low frequency):** Slow variation along the sequence (candidate long-range structure).
- $k = 6$–20 **(mid frequency):** Medium-scale variation (candidate local transitions).
- $k > 20$ **(high frequency):** Rapid variation (candidate token-level effects and noise).

4.2 Why DCT on Embeddings Works

The DCT-II is the cosine expansion associated with an even boundary extension of a finite sequence. Padding makes token sequences finite, but it does not imply that they are smooth or that the DCT is optimal. Moreover, applying the transform to zero-padded sequences can encode sequence length and boundary artifacts. A valid benchmark must therefore transform only valid-token positions or

otherwise mask and rescale padding, and it must include a length-only control. Energy compaction is a property to measure, not assume.

Regularity heuristic. If each embedding coordinate is well approximated by a sufficiently regular function of position and the boundary extension is compatible with that regularity, classical Fourier/cosine approximation results predict decaying high-frequency coefficients. The rate depends on the precise smoothness and boundary assumptions. We do not assume, or establish, a universal $O(k^{-2})$ rate for pretrained-transformer embeddings.

Several mechanisms could create low-frequency structure—local linguistic coherence, contextual mixing, and positional encoding—but none guarantees coordinate-wise smoothness. The experiments therefore test rather than presuppose the relevant spectral concentration.

Whether most task-relevant information is concentrated in the first $K \ll L$ modes is therefore an empirical question evaluated by truncation and downstream-prediction experiments.

Related architectural evidence from FNet. Lee-Thorp et al. (2022) found that parameter-free Fourier token mixing could remain competitive on several GLUE tasks while reducing training cost. FNet concerns token mixing inside a trained architecture; it motivates spectral investigation but does not establish DCT coefficient decay or the semantic sufficiency of our extracted features.

Connection to positional encodings. The original Transformer (Vaswani et al., 2017) uses sinusoidal positional encodings $\mathrm{PE}(pos, 2i) = \sin(pos/10000^{2i/d})$. This makes a frequency-domain analysis natural, but the DCT of contextual embeddings is not an inverse of the positional-encoding map and should not be interpreted as one.

4.3 Pipeline and Features

The Spectral Text pipeline:

$$\text{text} \xrightarrow{\text{tokenize}} \text{tokens} \xrightarrow{\text{embed}} \mathbf{E} \in \mathbb{R}^{L\times d} \xrightarrow{\text{DCT}} \hat{\mathbf{E}} \in \mathbb{R}^{K\times d} \xrightarrow{\text{features}} \mathbf{z} \xrightarrow{\text{Ridge}} \hat{y}$$

Feature extraction from the truncated DCT coefficient matrix $\hat{\mathbf{E}} \in \mathbb{R}^{K\times d}$:

1. **Flattened coefficients:** $\mathrm{vec}(\hat{\mathbf{E}}) \in \mathbb{R}^{Kd}$ — the full spectral representation.
2. **Energy profile:** $e_k = \|\hat{\mathbf{E}}_{k,:}\|_2$ for $k = 0, \ldots, K-1$ — how much energy each frequency band carries. For dimensionality reduction, the current implementation applies fold-local SVD to the flattened coefficients. It emits flattened coefficients and per-mode energy; cumulative-energy summaries are not part of the implemented feature vector. The final feature vector optionally includes the standard sentence embedding for comparison.

4.4 Leakage-Safe Benchmark Protocol

The text case is evaluated as a diagnostic rather than treated as established evidence. The intended datasets are Rotten Tomatoes and SST-2 with 5,000 examples each, using the frozen all−MiniLM−L6−v2 encoder. The executable benchmark currently evaluates Ridge on sentence embeddings, valid-token DCT features, and fold-local SVD/LDA projections. A complete future comparison must add declared nonlinear baselines such as HistGradientBoosting and MLPs before the nonlinear-gap hypothesis can be tested.

The outer evaluation uses five stratified folds with a fixed split seed. Every data-dependent transformation—including SVD, PCA, LDA, scaling, and hyperparameter selection—must be fit using only the outer training fold. DCT features must be computed from valid-token positions, with sequence length reported as a separate control; padded positions must not silently determine the spectrum. The test fold is used once for evaluation. The result bundle records the script hash, cache manifest, protocol, feature counts, fold-level scores, and aggregate metrics.

4.5 Current Reproducibility Status

An earlier exploratory pipeline fit some SVD/PCA transforms on the full dataset and one LDA transform on all labels before cross-validation. Those results are not admissible evidence because the preprocessing was transductive and the LDA path leaked targets. We therefore withdraw the associated accuracy tables and all derived claims from the evidential core of this version.

The corrected Ridge benchmark is implemented in tools/spectral_text/spectral_v2_benchmark.py. Regression tests verify that changing test-fold observations cannot change the training-fold PCA features, changing test labels cannot change LDA features, padded token values cannot change valid-token DCT features, legacy caches without valid lengths are rejected, and legacy full-dataset PCA entry points fail closed. The implementation does not yet contain the prospective nonlinear comparison set. At the time of this revision, the required embedding caches are absent from the local reproducibility surface. The machine-readable result bundle consequently records blocked_cache_unavailable_or_incompatible and contains no replacement accuracy claims.

4.6 Falsifiable Text Hypotheses

Once the caches are regenerated, the current Ridge benchmark can test the first two hypotheses:

1. Low-order DCT features retain predictive signal beyond a sentence-embedding-only baseline.
2. A spectral composite without sentence pooling retains a substantial fraction of baseline accuracy. The third hypothesis—that the gap between Ridge and tuned nonlinear baselines is small on the strongest representation after controlling for sequence length—requires implementation and testing of the declared nonlinear comparison set before it is executable.

These are hypotheses, not conclusions of the current paper version. A future numerical revision must report fold-level scores and preprocessing provenance before any of them can be promoted to findings.

---

5. Discussion

5.1 The SLB Design Recipe

The framework suggests a recipe for any new domain:

1. **Identify the function class.** What mathematical structure does the target have? Smooth? Threshold-heavy? Periodic? Sequential? Hierarchical?
2. **Select candidate dictionaries.** Use domain theory to propose representations with favorable approximation or invariance properties.
3. **Extract features.** Apply the basis transform to the raw data.
4. **Fit Ridge.** Use Ridge regression (with cross-validated α) on the basis features.

5. **Optionally add a residual model.** If the basis captures most but not all of the target, use a flexible model (XGBoost, neural network) on the residual.

This recipe motivates:

- **Tabular:** analytic-residual-plus-jump hypothesis → cosine + sigmoid dictionary → Ridge → optionally a residual model.
- **Text:** positional-frequency hypothesis → DCT features → leakage-safe comparison of linear and nonlinear heads.

5.2 What About Other Modalities?

The SLB principle suggests candidate analyses in modalities where relevant structure can be characterized:

Modality	Candidate basis	Rationale
Time series	Wavelets, DCT	Multi-scale temporal structure
Spatial data (geo)	Spherical harmonics	Smooth functions on S^2
Graph features	Graph Fourier modes (eigenvectors of Laplacian)	Smooth functions on graphs (Shuman et al., 2013)
Audio	Mel-spectrogram + DCT (MFCCs)	Standard in speech recognition — already an SLB instantiation

MFCCs (Mel-Frequency Cepstral Coefficients) are the DCT of a log Mel-spectrum and provide a historical example of a structured transform paired with comparatively simple downstream models. They support the usefulness of the design pattern without establishing that one basis or one linear model is universally best for speech.

5.3 Relationship to Deep Learning

Deep neural networks learn feature maps implicitly. The hidden layers compute $\phi_\theta(\mathbf{x})$, and a linear final layer computes $\mathbf{w}^\top \phi_\theta(\mathbf{x})$. One SLB route replaces ϕ_θ with an explicit structured dictionary; another analyzes or meta-learns the representation while keeping the task-specific head simple.

This has trade-offs:

- **Constructed dictionaries can help in small samples:** Fewer representation parameters must be estimated from the task data, although the resulting estimator can still have high variance or poor conditioning.
- **Explicit features can aid interpretation:** Each $w_k b_k(\mathbf{x})$ is named, but correlated or redundant dictionaries can make coefficient-level interpretations unstable.
- **Learned representations help when useful structure is unknown:** They can discover task-relevant features, but success is empirical rather than guaranteed by gradient descent alone.
- **Scale changes the trade-off:** Large pretrained models can amortize representation learning over extensive data and many tasks.

The two approaches are complementary. The SLB perspective can inform neural architecture design: if the target has plausible spectral structure, DCT or cosine layers provide a testable inductive bias. Whether that bias accelerates learning must be established empirically.

5.4 Limitations

- **Text evidence is pending replication.** The corrected Ridge path is implemented, but the required embedding caches are absent and the prospective nonlinear baselines are not yet implemented. No text accuracy or nonlinear-gap claim is treated as established in this version.
- **An invertible DCT adds no information by itself.** Truncation, pooling, supervised projections, and combinations can change what information is retained and how easily a downstream estimator accesses it; these effects must be measured fold-locally.
- **Embedding quality is assumed.** The text instantiation depends on a pretrained transformer for token embeddings. The DCT is applied to the *output* of a neural network, not to raw text. The SLB framework's text contribution is characterizing the structure of embeddings, not replacing the embedding model.
- **Basis selection is manual.** The framework does not automate basis selection — the practitioner must identify the function class. Automatic basis discovery (learning the basis structure from data) remains open.
- **Cross-modal compositionality.** For multimodal inputs (text + tabular + image), how to combine per-modality SLB bases is unexplored.
- **Planned text scope is narrow.** The current protocol targets Rotten Tomatoes and SST-2. Broader datasets and multiclass tasks remain outside this version.

---

6. Learned Bases as a Companion Direction

When a suitable feature family is unknown, it is natural to learn a map $\psi_\theta : \mathcal{X} \to \mathbb{R}^K$ from a distribution of tasks and fit a task-specific Ridge head on the support set. This mechanism is not new: R2-D2 meta-trains a feature extractor through a differentiable closed-form Ridge solver (Bertinetto et al., 2019). Related analyses show that much of few-shot meta-learning performance can reside in the learned representation and a simple last-layer adaptation rule (Goldblum et al., 2020).

The companion manuscript *Meta-SLB: Learning the Basis, Not the Weights* studies regression and theorem-proving applications of this mechanism. It must be read as an application and reframing of differentiable last-layer meta-learning, not as the invention of closed-form Ridge adaptation. Its benchmark claims, baselines, and computational provenance require their own publication passport and are not evidence used by the present paper.

For the SLB framework, the relevant conceptual point is modest: explicit basis construction and meta-learned feature construction are alternative ways to relocate complexity from the task-specific predictor into the representation. Whether this relocation improves generalization is task- and protocol-dependent.

---

7. Linear-Probe Sufficiency on Pre-trained Representations

7.1 Diagnostic Hypothesis

If a representation makes a target approximately linear, a well-regularized linear probe should approach the performance of stronger downstream models. We operationalize this as a diagnostic gap between Ridge and a specified set of nonlinear baselines. A small observed gap establishes neither global optimality nor a property of all downstream tasks.

7.2 Experimental Design

The protocol compares three representation settings:

- **Raw features:** datasets on which nonlinear decision boundaries are plausible.
- **Pretrained embeddings:** sentence, DCT, and composite features from a frozen encoder.
- **Linear projections:** dimension-reduced embeddings used to separate information loss from downstream nonlinearity.

The intended model set contains Ridge, logistic regression, k-nearest neighbours, HistGradient-Boosting, and two MLP sizes. Only the Ridge path is implemented in the current benchmark. A future nonlinear-gap result must implement the remaining models, freeze explicit comparable tuning budgets, and use five-fold stratified outer cross-validation with all preprocessing and model selection nested inside each training fold.

7.3 Diagnostic Definition

For representation ϕ, dataset D, linear model class $\mathcal{L}$, and declared nonlinear comparison set $\mathcal{M}$, define

$$\Delta(\phi, D; \mathcal{L}, \mathcal{M}) = \max_{m \in \mathcal{M}} \widehat{S}(m \circ \phi; D) - \max_{\ell \in \mathcal{L}} \widehat{S}(\ell \circ \phi; D),$$

where $\widehat{S}$ is an outer-fold performance estimate with higher values better. A small Δ means only that the declared nonlinear set did not materially improve on the declared linear set under that protocol. It is not invariant to model choice, tuning budget, sample size, or dataset.

7.4 Current Status

The earlier numerical ladder and average-gap table shared the exploratory preprocessing provenance described in §4.5 and are withdrawn from the evidential core. They will be reinstated only after the leakage-safe pipeline produces a complete result bundle with fold-level scores. Until then, this section defines a diagnostic rather than reports a finding.

Linear probing remains a natural measurement tool for the SLB perspective. The framework organizes the question; it does not claim priority for linear probing or imply that all useful representations yield linear targets.

8. Unified View

The framework has three logically distinct components:

Principle	Evidence	Section
Fixed representation → unique regularized linear solution	Proposition 1	§2
Analytic residual + jumps → explicit approximation rate	Proposition 2	§3
Representation quality → measurable linear/nonlinear gap	Leakage-safe diagnostic protocol	§§4, 7

The SLB framework is not just a perspective on feature engineering — it offers basis/feature construction as a useful organizing principle for the settings studied here. Gradient descent, trees, and depth can be viewed as alternative mechanisms for arriving at effective representations, but we do not claim a universal characterization of all machine learning as "basis construction."

9. Conclusion

The Structured Latent Basis perspective separates two questions that are often conflated: whether a representation makes a task simple, and how to fit a predictor once that representation is fixed. Proposition 1 resolves only the second question for the Ridge objective. Proposition 2 gives one restricted setting in which cosine and sigmoid features control approximation error. Neither result says that an adequate low-dimensional representation exists for every learning problem.

The empirical consequence is a disciplined diagnostic. Propose a structured representation, fit all learned preprocessing within the training fold, and measure the gap between a regularized linear probe and declared nonlinear baselines under comparable tuning budgets. The text implementation now enforces that protocol, but its missing caches prevent numerical conclusions in this version. This boundary is intentional: the framework remains useful only if representation claims are separated from leakage, baseline weakness, and post hoc interpretation.

Declaration of Generative AI and AI-assisted technologies in the writing process

During the preparation of this work the author used Cursor and its integrated AI models (Claude and GPT families) in order to assist with manuscript drafting, structural editing, and language polishing. After using this tool/service, the author reviewed and edited the content as needed and takes full responsibility for the content of the publication.

References

- Almarwani, N., Aldarmaki, H., and Diab, M. (2019). Efficient Sentence Embedding using Discrete Cosine Transform. *Proceedings of EMNLP-IJCNLP 2019*, 3672–3678. DOI: 10.18653/v1/D19-1380.
- Bertinetto, L., Henriques, J. F., Torr, P. H. S., and Vedaldi, A. (2019). Meta-learning with differentiable closed-form solvers. *International Conference on Learning Representations*. https://openreview.net/forum?id=HyxnZh0ct7.
- Chen, T., and Guestrin, C. (2016). XGBoost: A scalable tree boosting system. *Proceedings of KDD 2016*, 785–794. DOI: 10.1145/2939672.2939785.
- Gangeh, M. J., Farahat, A. K., Ghodsi, A., and Kamel, M. S. (2015). Supervised Dictionary Learning and Sparse Representation—A Review. arXiv:1502.05928. https://arxiv.org/abs/1502.05928.
- Goldblum, M., Reich, S., Fowl, L., Ni, R., Cherepanova, V., and Goldstein, T. (2020). Unraveling Meta-Learning: Understanding Feature Representations for Few-Shot Tasks. *Proceedings of ICML 2020*, 3607–3616. https://proceedings.mlr.press/v119/goldblum20a.html.
- Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (2nd ed.). Springer. DOI: 10.1007/978-0-387-84858-7.
- Lee-Thorp, J., Ainslie, J., Eckstein, I., and Ontañón, S. (2022). FNet: Mixing Tokens with Fourier Transforms. *Proceedings of NAACL-HLT 2022*, 4296–4313. DOI: 10.18653/v1/2022.naacl-main.319.
- Mairal, J., Bach, F., Ponce, J., Sapiro, G., and Zisserman, A. (2008). Supervised Dictionary Learning. *Advances in Neural Information Processing Systems 21*. https://arxiv.org/abs/0809.3083.
- Müller-Eberstein, M., van der Goot, R., and Plank, B. (2022). Spectral Probing. *Proceedings of EMNLP 2022*, 7730–7741. DOI: 10.18653/v1/2022.emnlp-main.527.
- Nagy, T. (2026). The Smooth-Step Spectral Method: Unifying Smooth and Threshold Structure in Tabular Regression. *Working paper*.
- Rahimi, A., and Recht, B. (2007). Random Features for Large-Scale Kernel Machines. *Advances in Neural Information Processing Systems 20*, 1177–1184.
- Reimers, N., and Gurevych, I. (2019). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. *Proceedings of EMNLP-IJCNLP 2019*, 3982–3992. DOI: 10.18653/v1/D19-1410.
- Shuman, D. I., Narang, S. K., Frossard, P., Ortega, A., and Vandergheynst, P. (2013). The Emerging Field of Signal Processing on Graphs: Extending High-Dimensional Data Analysis to Networks and Other Irregular Domains. *IEEE Signal Processing Magazine*, 30(3), 83–98. DOI: 10.1109/MSP.2012.2235192.
- Trefethen, L. N. (2013). *Approximation Theory and Approximation Practice*. SIAM. DOI: 10.1137/1.9781611975949.
- Vaswani, A., Shazeer, N., Parmar, N., et al. (2017). Attention Is All You Need. *Advances in Neural Information Processing Systems 30*.

Appendix A: Text Benchmark Details

A.1 Embedding Model

We use all−MiniLM−L6−v2 (Reimers & Gurevych, 2019), a 6-layer distilled BERT model producing 384-dimensional token and sentence embeddings. Token sequences are truncated to at most 128 positions. Publication-grade DCT features must use valid-token lengths rather than treating zero padding as signal, and the benchmark must include a length-only control.

A.2 DCT Configuration

- DCT-II applied along axis 0 (position), per embedding dimension.
- Truncation to $K \in \{8, 16\}$ frequency modes in the current protocol.
- Implemented feature types: flattened coefficients ($K \times d$) and energy profile (K).
- SVD/PCA compression: the component count is bounded by the training-fold sample size and feature dimension; each transform is fit separately inside the outer training fold.

A.3 Classification Model

RidgeClassifierCV with $\alpha \in \{0.01, 0.1, 1, 10, 100, 1000\}$ is fit inside each outer training fold. The pretrained encoder remains frozen. HistGradientBoosting, MLP, logistic-regression, and k-nearest-neighbour baselines are prospective requirements, not components of the current script; when added, they must use declared search spaces and the same outer folds.

A.4 Reproducibility

Embeddings are generated by precompute_v2.py; evaluation is run by spectral_v2_benchmark.py. The cache records valid-token lengths, and the benchmark rejects legacy cache files that omit them. The benchmark emits a JSON bundle containing its script hash, cache manifest, protocol, feature counts, fold-level scores, and aggregates. All split and decomposition random states are fixed at 42. The current bundle has status blocked_cache_unavailable_or_incompatible, so it contains no accepted numerical results.

A.5 Result Admission Rule

A numerical table may enter this paper only when the result bundle has status complete, both requested cache records are present, every fitted transform is fold-local, and fold-level scores are available for each reported aggregate. Exploratory results produced by the retired global-preprocessing pipeline remain development history and are not publication evidence.

A.6 Regression Checks

tests/test_spectral_text_v2.py checks six invariants: test observations cannot alter training-fold PCA features; test labels cannot alter fold-local LDA features; padded values cannot alter valid-token DCT features; legacy caches without valid lengths are rejected; legacy global-PCA entry points fail closed; and JSON outputs record the protocol and completion status. These tests validate data separation and padding discipline in the benchmark implementation, not the scientific hypotheses themselves.