Calibrated Principal Component Regression

Yixuan Florence Wu
Northwestern University

Yilun Zhu
University of Michigan

Lei Cao
Northwestern University

Naichen Shi
Northwestern University

Abstract

We propose a new method for statistical inference in generalized linear models. In the overparameterized regime, Principal Component Regression (PCR) reduces variance by projecting high-dimensional data to a low-dimensional principal subspace before fitting. However, PCR incurs truncation bias whenever the true regression vector has mass outside the retained principal components (PC).

To mitigate the bias, we propose Calibrated Principal Component Regression (CPCR), which first learns a low-variance prior in the PC subspace and then calibrates the model in the original feature space via a centered Tikhonov step. CPCR leverages cross-fitting and controls the truncation bias by softening PCR's hard cutoff. Theoretically, we calculate the out-of-sample risk in the random matrix regime, which shows that CPCR outperforms standard PCR when the regression signal has non-negligible components in low-variance directions. Empirically, CPCR consistently improves prediction across multiple overparameterized problems. The results highlight CPCR's stability and flexibility in modern overparameterized settings.

1 INTRODUCTION

In this paper, we consider the problem of overparameterized regression, where the number of variables exceeds the number of samples. Overparameterization has become commonplace across scientific and engineering domains. Advances in data acquisition technologies routinely generate high-dimensional data with rich structure. For instance, functional MRI (fMRI) studies may record thousands of features for only a few hundred subjects (Casey et al., 2018), and single-cell RNA sequencing can measure expression levels of 10^3–10^5 genes across a comparable number of cells (Wu and Zhang, 2020). On the other hand, the development of foundation models, such as large language models (LLMs) (Radford et al., 2021) and vision foundation models (Siméoni et al., 2025), produces rich embeddings in thousands of dimensions that capture complex semantics and enable downstream tasks such as few-shot classification, inference, and reasoning (Yuan et al., 2021; Yu et al., 2022). Together, these trends have created a new data landscape where having far more features than samples is increasingly common.

Building statistical models in the overparameterized regime is not easy. From an information-theoretic standpoint, regression in such regimes is inherently ill-posed, where infinitely many solutions can interpolate the training data (McRae et al., 2022; Tsigler and Bartlett, 2023; Medvedev et al., 2024), raising concerns about identifiability and generalization.

Classical statistics resolves this ambiguity by imposing prior assumptions or structural constraints. Two canonical examples are LASSO (Tibshirani, 1996) and ridge regression (Hoerl and Kennard, 1970), which employ an ℓ_1 and an ℓ_2 penalty, respectively. Beyond explicit regularization, recent work has revealed that optimization dynamics can induce implicit regularization. For example, early stopping approximates ridge regression (Ali et al., 2019), and gradient descent in overparameterized models induces an implicit bias toward minimum-norm or low-complexity solutions (Hastie et al., 2022).

From a Bayesian perspective, however, these methods correspond to simple priors that assume isotropic feature distributions and overlook the heteroskedasticity of real data. An alternative line of work focuses on learning low-dimensional latent representations of the covariates to enable a range of downstream analyses (Wu and Zhang, 2020; Maulik et al., 2021; Recanatesi et al., 2021; Zhang et al., 2024; Bonneville et al., 2024). A classical statistical method along these lines is principal

Proceedings of the 29th International Conference on Artificial Intelligence and Statistics (AISTATS) 2026, Tangier, Morocco. PMLR: Volume 300.

component regression (PCR), which combines unsupervised dimension reduction with regression by projecting the data onto its leading principal components prior to fitting the model. Implicitly, PCR assumes that a small number of latent factors govern the variability of both covariates and responses, allowing it to identify the relevant low-dimensional structure prior to regression.

However, PCR suffers from a fundamental limitation that it decouples feature extraction from prediction. This decoupling is problematic when the regression signal does not align with the leading principal components of the covariates. In such cases, the projections of regression coefficients on directions with low variance in covariates are discarded, which introduces **truncation bias**. As a result, PCR may substantially degrade.

To mitigate truncation bias while still leveraging latent structure, we propose *Calibrated Principal Component Regression* (CPCR). CPCR introduces a sample-splitting and cross-fitting strategy: one split is used to learn a low-variance but potentially biased prior, and the other split calibrates regression coefficients in the full feature space. This two-stage design effectively corrects for bias introduced by discarding residual directions.

Figure 1 illustrates this scenario in a 2D classification setting where the label signal is not perfectly aligned with the top principal component, and CPCR achieves a more accurate decision boundary than PCR.

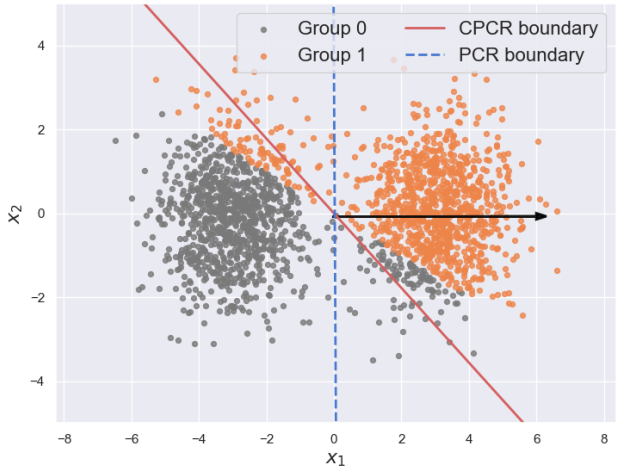

Figure 1: Data points with different membership, group 0 (grey dots) and group 1 (orange dots). The black arrow shows the top principal component. When considering the top PC only, PCR's decision boundary is influenced by truncation bias, while CPCR is able to calibrate its decision boundary.

Is the superior performance of CPCR merely an artifact of the specific dataset? Under what conditions does calibration provably mitigate truncation bias? To address these questions, we provide *exact* theoretical characterizations of the out-of-sample prediction risk of CPCR under least-squares regression, leveraging tools from random matrix theory.

We support these theoretical findings with extensive numerical experiments. The results consistently demonstrate that CPCR outperforms PCR, highlighting the benefit of calibration in reducing truncation bias. Moreover, CPCR is superior to ridge regression when the ground-truth regression vector is highly anisotropic and well-aligned with the principal subspace.

On a wide range of overparameterized regression and classification problems, we evaluate CPCR with generalized linear models. Our results show consistent improvement over existing popular methods in high-dimensional regression.

In summary, our main contributions are:

1. We present a novel algorithm, CPCR, that leverages cross-fitting to exploit the low-dimensional structure in covariate while controlling the truncation bias.

2. We develop new analytical tools to derive the *exact* asymptotic risk of CPCR using random matrix theory. Our analysis quantifies how risk depends on (i) the alignment of the regression coefficient with the informative subspace and (ii) the spectral distribution of the covariance matrix constrained on the orthogonal subspace of principal components.

3. We validate our theoretical risk characterization through synthetic experiments and demonstrate that CPCR consistently outperforms PCR and other popular baselines on both synthetic and real-world datasets.

1.1 Related Works

We review several lines of related research most relevant to our framework of low-rank regression and calibration.

Principal component regression (PCR). PCR builds a regression model after projecting covariates onto leading principal components of the sample covariance. A recent line of work provides asymptotic and high-dimensional risk characterizations for PCR when the principal components are estimated from data rather than known a priori. In particular, Gedon et al. (2024) provide asymptotic guarantees for generalization risk when data is sampled from a spiked covariance model, where the population covariance is

nearly isotropic except for a few low-rank signal directions with elevated eigenvalues. Under a similar setting, Green and Romanov (2024) not only characterizes the exact asymptotic risk of PCR but also reveals the dependencies of optimal number of PCs. Agarwal et al. (2023) propose an online variant of PCR with finite sample guarantees. Similar to ours, Agarwal et al. (2019) also considers the overparameterized regime. They study the robustness of PCR to noise, sparsity, and mixtures of the covariates. These works establish a solid theoretical foundations for PCR. Our contribution extends prior analyses by introducing a calibration step to standard PCR that mitigates its potential bias.

Ridge regression in high dimensions. Ridge regression provides an isotropic prior to the regression coefficient by shrinking coefficients uniformly along all directions. Bartlett et al. (2020) give necessary and sufficient spectral conditions under which ridgeless regression achieves nearly optimal risk in overparameterized linear models, thereby establishing when benign overfitting occurs. Hastie et al. (2022) characterize the generalization risk of ridge and ridgeless regression under isotropic and anisotropic features, and even for data transformed by a one-layer random neural network. Bach (2023) compute asymptotic equivalents for generalization performance when the data are first passed through a random projection matrix. More recently, Cheng and Montanari (2025) establish non-asymptotic bias and variance bounds that hold beyond proportional asymptotics. Our theoretical analysis differs in two important ways. First, CPCR introduces a calibration step that solves a centered, regularized optimization problem, rather than relying on global isotropic shrinkage. Second, our theoretical analysis of asymptotic generalization risk extends beyond classical ridge regression by explicitly incorporating sample splitting and cross-fitting, which alters both the bias–variance tradeoff and the dependence on feature covariance structure.

Debiased regression and machine learning. Chernozhukov et al. (2018) introduce a general framework that combines orthogonalization with cross-fitting to mitigate regularization bias when using machine learning in high-dimensional settings. To debias linear regression, Yi and Neykov (2021) propose a split-sample procedure, where the first half of the data is used to obtain a regularized estimator, and the second half is used to construct a debiased version without cross-fitting. Their abstract framework encompasses important classes of estimators, including the group LASSO and the Sorted L-One Penalized Estimator (SLOPE). Our debiasing scheme follows a similar split-then-debias principle, though specialized to principal component regression. In a broader perspective, debiasing has also been applied in other areas of machine learning, including semi-supervised learning, Gaussian processes, and prediction-powered inference (Schmutz et al., 2023; Hu et al., 2022; Ji et al., 2025). CPCR can be viewed as a specialized instance of debiased or calibrated estimation. Crucially, the structure of our problem allows us to derive stronger theoretical guarantees compared to general debiased machine learning frameworks.

Supervised dimension reduction. Partial least squares (PLS) and canonical-correlation analysis (CCA) are classical supervised linear methods that maximize covariance or correlation between projections of covariates and response. While CCA relies on inverting sample covariance matrices, which makes it unstable in high-dimensional regimes, PLS avoids this via iterative algorithms (e.g., NIPALS (Wold, 1975)). However, since both methods intrinsically rely on a small number of components, they are also subject to truncation bias. Sliced Inverse Regression (SIR) (Li, 1991) and Supervised PCA (Bair et al., 2006; Ritchie et al., 2020) construct supervised low-dimensional embeddings of covariates using nonlinear or semi-parametric transformations. More specifically, SIR identifies directions that explain how the inverse regression changes across slices of response, which estimates a low-dimensional subspace of the predictors relevant for predicting response. However, the standard form of SIR is known to be brittle in the overparameterized regime (Li and Yin, 2008). On the other hand, supervised PCA finds principal components by weighting each feature according to its covariance with response. In contrast to these supervised approaches, CPCR builds on the unsupervised framework of PCR, introducing a calibration step within the regression stage rather than constructing response-dependent embeddings of the predictors.

2 PROBLEM FORMULATION

Our goal is to learn a predictive model that maps high-dimensional covariates to responses. Let $\boldsymbol{X} \in \mathbb{R}^{p \times n}$ denote the data matrix, where p is the number of features and n is the number of samples, and let $\boldsymbol{y}$ denote the n-dimensional response vector whose entries are continuous for regression tasks and categorical for classification tasks. We focus on the high-dimensional regime where p is comparable to or larger than n.

We model the conditional distribution of the response $\boldsymbol{y}$ given the covariates $\boldsymbol{X}$ using a generalized linear

model (GLM):

$$p(\boldsymbol{y} \mid \boldsymbol{X}) = \rho\big(\boldsymbol{y};\, \boldsymbol{X}^\top \boldsymbol{\gamma}\big)\,, \tag{1}$$

where $\boldsymbol{\gamma} \in \mathbb{R}^p$ is the regression coefficient vector, and ρ denotes a density or mass function from the exponential family parameterized by the linear predictor $\boldsymbol{X}^\top \boldsymbol{\gamma}$. Standard choices of ρ include Gaussian and Bernoulli distributions, corresponding to least-squares linear and logistic regression, respectively.

We further consider the intrinsic structure of $\boldsymbol{X}$. A natural assumption is that the predictive signal in $\boldsymbol{X}$ can be well-approximated by a rank-r subspace $\boldsymbol{U} \in \mathbb{R}^{p\times r}$, with $r \ll p$. Exploiting such low-rank structure is especially valuable in the overparameterized regime, where direct estimation in the ambient space suffers from high variance and instability.

To quantify the alignment between the regression coefficient vector $\boldsymbol{\gamma}$ and the informative subspace, we introduce the following notion of a predictive-power coefficient.

Definition 1 *Given a coefficient vector $\boldsymbol{\gamma}$ and a low-rank subspace $\boldsymbol{U}$, the* predictive-power coefficient κ *is defined as*

$$\kappa \;=\; \frac{\|\boldsymbol{\Pi}_{\boldsymbol{U}}\boldsymbol{\gamma}\|_2^2}{\|\boldsymbol{\gamma}\|_2^2} \;\in\; (0,1), \tag{2}$$

where $\boldsymbol{\Pi}_{\boldsymbol{U}}$ is the projection operator onto column space of $\boldsymbol{U}$, and $\|\cdot\|_2$ denotes the Euclidean norm.

Intuitively, κ measures the fraction of the regression signal explained by the subspace $\boldsymbol{U}$. Values close to 1 indicate that the predictive signal is well captured by $\boldsymbol{U}$, while smaller values reflect stronger contributions from residual directions outside the subspace $\boldsymbol{U}$.

When κ is close to 1, a natural strategy is principal component regression (PCR). Suppose we have an estimate $\hat{\boldsymbol{U}} \in \mathbb{R}^{p\times r}$ that approximates the column space of $\boldsymbol{U}$. We project the data onto this subspace to obtain low-dimensional embeddings $\hat{\boldsymbol{U}}^\top \boldsymbol{X}$, and solve the regression problem through $\min_{\boldsymbol{\zeta}}\ \mathcal{L}\big(\boldsymbol{y},\, (\hat{\boldsymbol{U}}^\top \boldsymbol{X})^\top \boldsymbol{\zeta}\big)$, where $\mathcal{L}$ denotes the negative log-likelihood loss function associated with the GLM in (1). By projecting onto $\hat{\boldsymbol{U}}$, PCR reduces the effective degrees of freedom of the covariates, thereby reducing the variance of the estimates.

3 METHOD

In practice, the predictive-power coefficient κ is rarely equal to 1. Standard PCR therefore suffers from truncation bias, since the complementary component $(\boldsymbol{I}_p - \boldsymbol{\Pi}_{\boldsymbol{U}})\boldsymbol{\gamma}$ is completely discarded. When κ is not close to 1, ignoring these low-variance but informative directions can lead to substantial estimation error.

To address this challenge, we propose a *calibration-based* approach that combines dimension reduction with a debiasing step via sample splitting and cross-fitting. The procedure ensures that information in the discarded subspace is partially recovered, thereby mitigating truncation bias while retaining the variance reduction benefits of PCR.

More specifically, we first randomly split the labeled dataset $(\boldsymbol{X}, \boldsymbol{y})$ into two parts with equal size: $\boldsymbol{X} = \boldsymbol{X}_1 \cup \boldsymbol{X}_2$ and $\boldsymbol{y} = \boldsymbol{y}_1 \cup \boldsymbol{y}_2$. Next, we fit an estimator separately on each split and then combine them through calibration steps. We provide an explicit algorithm for computation described as below.

Step 1: PCR. On $(\boldsymbol{X}_1, \boldsymbol{y}_1)$, we fit a PCR model and obtain:

$$\hat{\boldsymbol{\zeta}}_1 = \arg\min_{\boldsymbol{\zeta}} \mathcal{L}(\boldsymbol{y}_1, (\hat{\boldsymbol{U}}^\top \boldsymbol{X}_1)^\top \boldsymbol{\zeta})\,. \tag{3}$$

Step 2: calibration. We construct $\boldsymbol{\gamma}_1^{\texttt{init}}$ as $\boldsymbol{\gamma}_1^{\texttt{init}} = \hat{\boldsymbol{U}}\hat{\boldsymbol{\zeta}}_1$, with $\hat{\boldsymbol{\zeta}}_1$ estimated from the first step. Then, on the held-out split $(\boldsymbol{X}_2, \boldsymbol{y}_2)$, we calibrate $\boldsymbol{\gamma}$ as

$$\hat{\boldsymbol{\gamma}}_1^{\texttt{calib}} = \arg\min_{\boldsymbol{\gamma}} \mathcal{L}(\boldsymbol{y}_2, \boldsymbol{X}_2^\top \boldsymbol{\gamma}) + \lambda \left\|\boldsymbol{\gamma} - \boldsymbol{\gamma}_1^{\texttt{init}}\right\|^2 \tag{4}$$

The calibration step prevents radically misspecified $\boldsymbol{\gamma}_1^{\texttt{init}}$ by regularizing it with the likelihood from the held-out set.

Step 3: exchange and re-calibration. Since we potentially only leverage the information from half of the samples in step 1 and 2, we exchange $(\boldsymbol{X}_1, \boldsymbol{y}_1)$ with $(\boldsymbol{X}_2, \boldsymbol{y}_2)$, and repeat step 1 and step 2 to obtain $\hat{\boldsymbol{\gamma}}_2^{\texttt{calib}}$. Finally, our estimator for CPCR is:

$$\hat{\boldsymbol{\gamma}}^{\texttt{CPCR}} = \frac{1}{2}\left(\hat{\boldsymbol{\gamma}}_1^{\texttt{calib}} + \hat{\boldsymbol{\gamma}}_2^{\texttt{calib}}\right). \tag{5}$$

The proposed CPCR estimator inherits the variance reduction benefits of PCR while correcting its truncation bias through calibration. The use of sample splitting and cross-fitting is essential in the overparameterized regime: without data splitting, PCR may interpolate the training set, leaving no residual signal for the calibration step to adjust, thereby rendering the procedure ineffective.

4 ANALYSIS OF CALIBRATION ESTIMATOR RISK

In this section, we develop the theoretical analysis of CPCR, providing an exact characterization of its

generalization risk and identifying conditions under which it excels.

Our theoretical analysis builds on recent advances in random matrix theory for high-dimensional regression (Dobriban and Wager, 2018; Hastie et al., 2022; Bach, 2023), which provide asymptotic characterizations of generalization risk for classical estimators such as ridge regression and PCR. Distinct from these settings, CPCR introduces a sample-splitting calibration step, leading to nontrivial dependencies between intermediate estimators $\hat{\gamma}_1^{\texttt{calib}}$ and $\hat{\gamma}_2^{\texttt{calib}}$. This structure gives rise to additional cross terms that are absent in standard analyses. We develop new analytical tools to characterize or estimate the cross terms by combining deterministic functional calculus, spectral asymptotics, and high-dimensional concentration arguments.

4.1 Risk Analysis of CPCR

For clarity, we dedicate our analysis of CPCR to the setting of least-squares regression, a representative case that enables analytical tractability, while many key insights about calibration and truncation bias are transferable to broader GLMs. In this case, the GLM in (1) specializes to the Gaussian linear model. Equivalently,

$$\boldsymbol{y} = \boldsymbol{X}^\top \boldsymbol{\gamma}^* + \boldsymbol{\epsilon}\,, \tag{6}$$

where $\boldsymbol{\epsilon} \in \mathbb{R}^{n\times 1}$ is a noise vector with independent entries with mean zero and variance σ^2. The analysis in this section is based on model (6).

Data generation process. We assume our covariate feature matrix $\boldsymbol{X}$ is generated by $\boldsymbol{X} = \boldsymbol{\Sigma}^{1/2}\boldsymbol{Z}$, with $\boldsymbol{Z} \in \mathbb{R}^{p\times n}$ having independent entries with zero mean and unit variance, and $\boldsymbol{\Sigma} \in \mathbb{R}^{p\times p}$ is a positive semidefinite population covariance matrix.

We further consider a spiked model, where the population covariance matrix $\boldsymbol{\Sigma} \in \mathbb{R}^{p\times p}$ can be written as

$$\boldsymbol{\Sigma} = \boldsymbol{U}\boldsymbol{\Sigma}_s\boldsymbol{U}^\top + \boldsymbol{V}\boldsymbol{\Sigma}_c\boldsymbol{V}^\top\,,$$

with entries of $\boldsymbol{\Sigma}_s \in \mathbb{R}^{r\times r}$ being low rank signals and $\boldsymbol{\Sigma}_c \in \mathbb{R}^{(p-r)\times(p-r)}$ represents the complementary background structure that is orthogonal to the signals space. Without loss of generality, both $\boldsymbol{\Sigma}_s$ and $\boldsymbol{\Sigma}_c$ are assumed to be diagonal. In addition, $\boldsymbol{U} \in \mathbb{R}^{p\times r}$ and $\boldsymbol{V} \in \mathbb{R}^{p\times(p-r)}$ are orthonormal factors with $\boldsymbol{U}^\top\boldsymbol{U} = \boldsymbol{I}_r$, $\boldsymbol{V}^\top\boldsymbol{V} = \boldsymbol{I}_{p-r}$ and $\boldsymbol{U}^\top\boldsymbol{V} = \boldsymbol{0}$.

Distributional assumptions on $\boldsymbol{\gamma}^*$. In many applications, predictive information is concentrated in a small number of dominant directions, while the remaining directions contribute little signal. To capture this structure, we develop a prior distribution of $\boldsymbol{\gamma}^*$ that is compatible with the definition of κ in (2).

$$\boldsymbol{\Sigma}_{\boldsymbol{\gamma}^*} = \kappa\boldsymbol{\Pi}_{\boldsymbol{U}} + (1-\kappa)\boldsymbol{\Pi}_{\boldsymbol{V}}\,. \tag{7}$$

One can verify that this modeling choice provides a versatile data generation process that aligns with the predictive-power coefficient κ.

Risk. Given an estimator $\hat{\boldsymbol{\gamma}}$ fitted from $(\boldsymbol{X}, \boldsymbol{y})$, we evaluate its performance on a new independent covariate $\boldsymbol{x}_0 = \boldsymbol{\Sigma}^{1/2}\boldsymbol{z}_0$ drawn from the population distribution. The out-of-sample prediction risk is defined as

$$\mathcal{R}(\hat{\boldsymbol{\gamma}}) \triangleq \mathbb{E}_{\boldsymbol{x}_0,\boldsymbol{\gamma}^*}[(\boldsymbol{x}_0^\top\hat{\boldsymbol{\gamma}} - \boldsymbol{x}_0^\top\boldsymbol{\gamma}^*)^2]\,. \tag{8}$$

Though $\hat{\boldsymbol{\gamma}}^{\texttt{CPCR}}$ is dependent on training set $(\boldsymbol{X}, \boldsymbol{y})$, whose risk is a random variable, the following theorem presents a deterministic limit of $\mathcal{R}(\hat{\boldsymbol{\gamma}}^{\texttt{CPCR}})$ in the large n and p regime.

Theorem 1 (Exact calibration estimator risk.) *In the asymptotic limit where $n, p \to \infty$ such that $p/n \to c \in (1,\infty)$. With high probability, the risk of our calibration estimator $\hat{\boldsymbol{\gamma}}^{\texttt{calib}}$ converge to a deterministic limit:*

$$\mathcal{R}(\hat{\boldsymbol{\gamma}}^{\texttt{CPCR}}) \to B_{\boldsymbol{X}}(\hat{\boldsymbol{\gamma}}^{\texttt{CPCR}}) + V(\hat{\boldsymbol{\gamma}}^{\texttt{CPCR}})\,, \tag{9}$$

where the bias term can be further estimated by

$$\begin{aligned} B_{\boldsymbol{X}}(\hat{\boldsymbol{\gamma}}^{\texttt{CPCR}}) =& \frac{1}{4}(B_{\boldsymbol{X}}(\hat{\boldsymbol{\gamma}}_1^{\texttt{calib}}) + B_{\boldsymbol{X}}(\hat{\boldsymbol{\gamma}}_2^{\texttt{calib}})) \\ &+ \frac{1}{2}B_{\boldsymbol{X}}(\hat{\boldsymbol{\gamma}}_1^{\texttt{calib}}, \hat{\boldsymbol{\gamma}}_2^{\texttt{calib}}) \end{aligned}$$

and the variance term can be estimated by

$$V(\hat{\boldsymbol{\gamma}}^{\texttt{CPCR}}) = \frac{1}{4}(V_{\epsilon_2}(\hat{\boldsymbol{\gamma}}_1^{\texttt{calib}}) + V_{\epsilon_1}(\hat{\boldsymbol{\gamma}}_2^{\texttt{calib}}))\,.$$

Moreover, consider $\{\mu_j(\boldsymbol{\Sigma}_c)\}_{j=1}^{p-r}$ the diagonal entries of $\boldsymbol{\Sigma}_c$ and $\nu_c = \frac{1}{p-r}\sum_{j=1}^{p-r}\delta_{\mu_j(\boldsymbol{\Sigma}_c)}$, then the almost sure limits of each individual terms are

$$\begin{aligned} &B_{\boldsymbol{X}}(\hat{\boldsymbol{\gamma}}_1^{\texttt{calib}}) \xrightarrow{a.s.} \\ &+(1-\kappa)(p-r)\int \frac{\mu}{1+\tilde{m}(-\lambda)\mu}d\nu_c(\mu) \\ &+(1-\kappa)(p-r)\tilde{m}_\rho(-\lambda)\int \frac{\mu}{(1+\tilde{m}(-\lambda)\mu)^2}d\nu_c(\mu) \\ &-(1-\kappa)(p-r)\tilde{m}(-\lambda)\int \frac{\mu^2}{(1+\tilde{m}(-\lambda)\mu)^2}d\nu_c(\mu)\,, \end{aligned} \tag{10}$$

$$\begin{aligned} &B_{\boldsymbol{X}}(\hat{\boldsymbol{\gamma}}_1^{\texttt{calib}}, \hat{\boldsymbol{\gamma}}_2^{\texttt{calib}}) \xrightarrow{a.s.} \\ &(1-\kappa)(p-r)\int \frac{\mu}{(1+\tilde{m}(-\lambda)\mu)^2}d\nu_c(\mu) \end{aligned} \tag{11}$$

and

$$V_{\epsilon_2}(\hat{\gamma}_1^{\texttt{calib}}) \xrightarrow{a.s.} \sigma^2 \left(\frac{\tilde{m}_z(-\lambda)}{\tilde{m}^2(-\lambda)} - 1 \right). \tag{12}$$

$\tilde{m}(\cdot)$ is defined implicitly as the solutions to a nonlinear equation depending on n, p and $\boldsymbol{\Sigma}$. $\tilde{m}_\rho(\cdot)$ and $\tilde{m}_z(\cdot)$ are partial derivatives with respect to ρ and z respectively.

From (9), two main observations arise: (i) the risk decreases as $\kappa \to 1$, and (ii) the limiting risk depends only on ν_c, the spectral distribution of the nuisance covariance $\boldsymbol{\Sigma}_c$, and not on the choice of basis $\boldsymbol{U}$ or $\boldsymbol{V}$. Based on (9), the following theorem specifies a condition for the optimal weight λ in (4).

Theorem 2 (Optimal λ.) *Under the same conditions as Theorem 1, the optimal lambda λ^* to achieve the minimal risk in (8) satisfies the following equation:*

$$B'_{\boldsymbol{X}}(\lambda^*) + V'(\lambda^*) = 0\,.$$

In particular,

$$\begin{aligned} B'_{\boldsymbol{X}}(\lambda) =& \frac{(1-\kappa)(p-r)}{2} \left(2 \int \frac{\mu^2 \tilde{m}_z(-\lambda)}{(1+\tilde{m}(-\lambda)\mu)^2} d\nu_c(\mu) \right. \\ & - \int \frac{\mu \tilde{m}_{\rho z}(-\lambda)}{(1+\tilde{m}(-\lambda)\mu)^2} d\nu_c(\mu) \\ & + 2 \int \frac{\mu^2 (\tilde{m}_\rho(-\lambda)\tilde{m}_z(-\lambda) + \tilde{m}_z(-\lambda))}{(1+\tilde{m}(-\lambda)\mu)^3} d\nu_c(\mu) \\ & \left. - 2 \int \frac{\mu^3 \tilde{m}(-\lambda)\tilde{m}_z(-\lambda)}{(1+\tilde{m}(-\lambda)\mu)^3} d\nu_c(\mu) \right) \end{aligned}$$

and

$$V'(\lambda) = \frac{\sigma^2}{2} \left(\frac{-\tilde{m}(-\lambda)\tilde{m}_{zz}(-\lambda) + 2\tilde{m}_z^2(-\lambda)}{\tilde{m}^3(-\lambda)} \right).$$

$\tilde{m}_{\rho z}(\cdot)$ denotes $\partial_{\rho z}\tilde{m}(\cdot)$ and $\tilde{m}_{zz}(\cdot)$ denotes $\partial_{zz}\tilde{m}(\cdot)$.

The proofs of Theorems 1 and 2, along with the explicit forms of $\tilde{m}(\cdot)$ and other auxiliary functions, are deferred to the Supplementary Materials. We will discuss the theoretical implications in the following sections.

4.2 Numerical Validations

Theorem 1 provides several key theoretical insights into the risk behavior of CPCR. Before discussing these in detail, we first conduct numerical experiments to verify the asymptotic predictions. For each choice of aspect ratio c and alignment parameter κ, we generate synthetic datasets following the process in Section 3, with $\boldsymbol{\gamma}^*$ sampled according to (7). Responses $\boldsymbol{y}$ are then generated using the linear model (6), yielding datasets $(\boldsymbol{X}, \boldsymbol{y})$. We fit CPCR on each dataset to obtain $\hat{\gamma}^{\texttt{CPCR}}$ and compute empirical risks using (8). We leverage the true signal subspace for both CPCR and PCR in this section. Each experiment is repeated 10 times to construct error bars. For comparison, we calculate the theoretical limits predicted by Theorem 1, where the optimal regularization parameter λ^* is given by Theorem 2. Figure 2 plots the empirical risks with error bars against the deterministic theoretical limits.

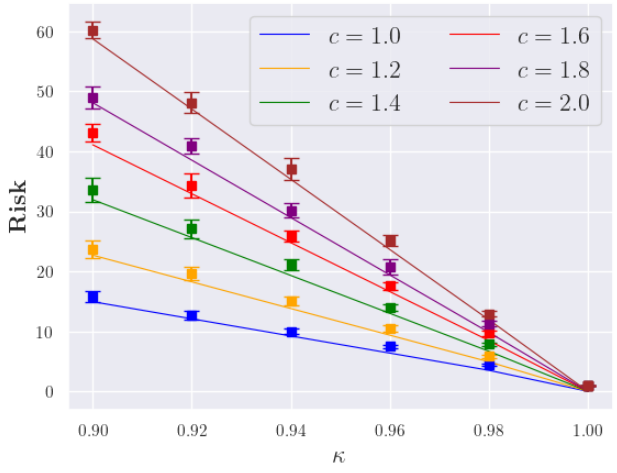

Figure 2: Theoretical limits of the empirical risks. Error bars denote the empirical risks (8) and solid curves denote the theoretical limits (9).

Figure 2 shows close agreement between empirical risks and their theoretical counterparts across a range of aspect ratios c. The tight concentration of the error bars around the curves confirms that our asymptotic analysis accurately characterizes even finite-sample behavior. Moreover, Figure 2 illustrates the monotone decrease in risk as κ increases. In particular, as κ approaches 1, the bias terms (10) and (11) completely vanish, leaving us with only the variance (12).

4.3 Implications of Theoretical Analysis

Risk dependence on spectral distributions. Since the bias term depends only on the eigenvalues of $\boldsymbol{\Sigma}_c$, we investigate how their magnitudes affect the risks. Specifically, we sample the entries of $\boldsymbol{\Sigma}_s$ from Unif$(2, 4)$ and those of $\boldsymbol{\Sigma}_c$ from Unif$(1, 3)$, where some eigenvalues of $\boldsymbol{\Sigma}_c$ are comparable to those of $\boldsymbol{\Sigma}_s$. In this regime, the signal and nuisance subspaces are not clearly separable (Figure 3). Several observations follow. First, CPCR consistently outperforms PCR except when κ is nearly 1, in which case they become comparable, which highlights the benefits of calibration. Second, both CPCR and PCR achieve low risk when κ is close to 1, indicating that they effectively leverage low-rank structure when the signals are concentrated

in $\boldsymbol{U}$. Interestingly, ridge regression performs worse in this regime, with substantially higher variance. This occurs probably because ridge enforces an isotropic prior that shrinks all directions uniformly, penalizing even the signal-aligned components of $\boldsymbol{\gamma}^*$ and thereby inflating both bias and variance. Third, as κ decreases, the risk of PCR grows rapidly due to truncation bias, whereas CPCR degrades more gracefully, which again confirms its robustness against truncation bias.

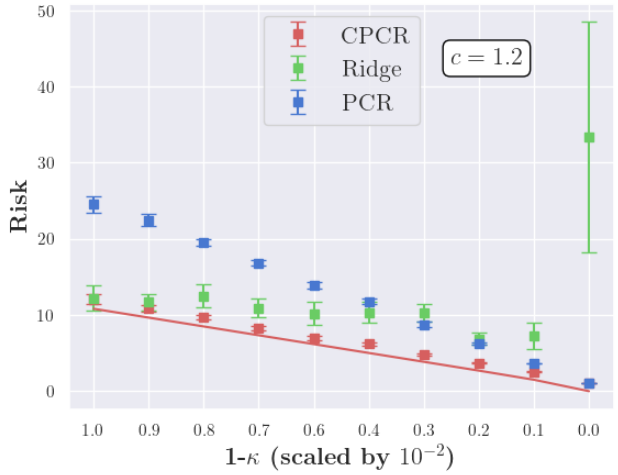

Figure 3: Risks of CPCR, PCR, and ridge regression. Eigenvalues of $\boldsymbol{\Sigma}_s$ are from $[2, 4]$ and those of $\boldsymbol{\Sigma}_c$ are from $[1, 3]$. CPCR consistently outperforms PCR and ridge regression.

To further examine the sensitivity of the risks to the eigenvalues of $\boldsymbol{\Sigma}_c$, we vary the uniform sampling range of its entries so that the mean spans values between 1 and 6, while fixing the variance at $\frac{1}{3}$. The entries of $\boldsymbol{\Sigma}_s$ are independently drawn from $\text{Unif}(2, 4)$, with mean 3 and variance $\frac{1}{3}$. As is evident from Figure 4, CPCR is substantially more robust to increasing background variance. Its risk increases at a slower rate than that of ridge regression or PCR as the background signal strength increases.

Optimal λ. As theorem 2 explains the relationship between the optimal calibration parameter λ^* and the predictive-power coefficient κ in a complicated manner, we plot λ^* under different settings of κ as the white curve in Figure 5.

In Figure 5, as κ approaches 1, λ^* increases dramatically, indicating that the initial PCR estimate is already well aligned with the true regression coefficient, and additional calibration offers only marginal improvements. Conversely, for smaller values of κ, the optimal λ^* becomes closer to zero. In such scenarios, the PCR solution does not offer strong guidance for estimating

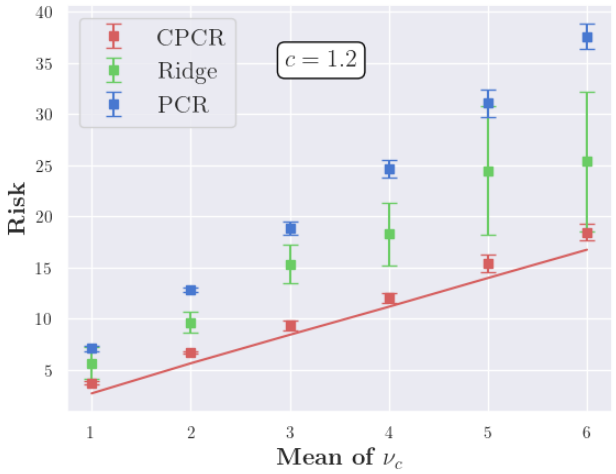

Figure 4: Risks of CPCR, PCR, and ridge regression with different range of eigenvalues in $\boldsymbol{\Sigma}_c$ with $\kappa = 0.995$. CPCR is more resistant to the increasing magnitudes of signals in the background space.

true regression coefficients, and the calibration step with a weaker regularization could significantly mitigate the truncation bias in the PCR solution. The behavior of optimal λ^* is consistent with our intuitions.

5 EXPERIMENTS

We present experimental results on synthetic and real datasets, showing CPCR's competitive or improved performance over benchmark methods across diverse tasks.

5.1 Synthetic Data

Data generation and methodology. We compare the performance of CPCR with ridge regression and PCR on synthetic data. Following the procedure in Section 3, we generate a feature matrix $\boldsymbol{X}$ where the eigenvalues of $\boldsymbol{\Sigma}_c \sim \text{Unif}(0, 1)$ are strictly smaller than those of $\boldsymbol{\Sigma}_s \sim \text{Unif}(2, 4)$, ensuring a clear separation between signal and noise directions. The regression coefficients $\boldsymbol{\gamma}^*$ are sampled according to (7), and responses $\boldsymbol{y}$ are generated from the linear model (6) with additive Gaussian noise. To introduce estimator mis-specification, we perform a singular value decomposition of $\boldsymbol{X}$ and deliberately retain fewer principal components than the true signal rank (Figure 6). PCR and CPCR are then applied in this reduced latent space, while ridge regression is fit in the original feature space. Empirical risks are evaluated via (8), and the experiment is repeated 10 times, with results summarized in box plots.

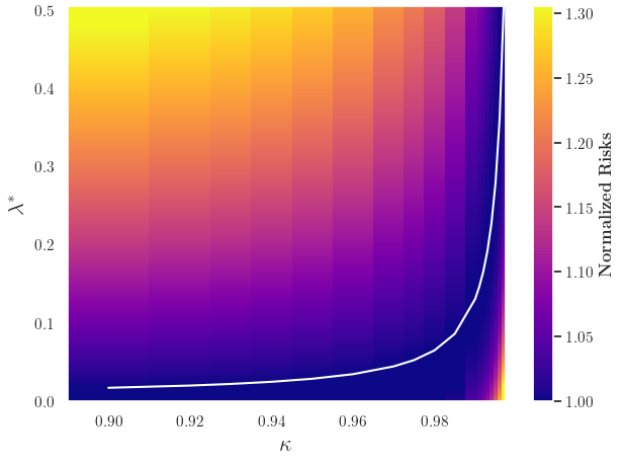

Figure 5: Normalized risk landscape over (κ, λ). The white curve traces the optimal regularization parameter λ^* for each κ. The background color represents the risk evaluated at each (κ, λ) pair, devided by the optimal risk corresponding to that κ. Darker (purple) regions indicate risks closer to the optimum, while brighter (yellow) regions correspond to higher risks.

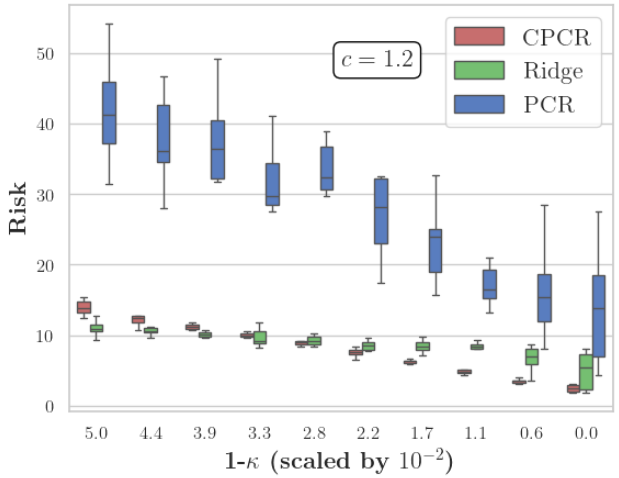

Figure 6: Risks of CPCR, ridge regression, and PCR when $\text{rank}(\boldsymbol{U}) = 10$ and $\text{rank}(\hat{\boldsymbol{U}}) = 5$. CPCR consistently outperforms PCR.

Results. It is evident that PCR suffers more severely from truncation bias when $\hat{\boldsymbol{U}}$ underestimates the rank, while CPCR is capable of calibrating on the misspecified subspace and successfully controlling that bias (Figure 6). In addition, CPCR also achieves superior or comparable performance with ridge regression when κ approaches 1. Our results demonstrate the robustness of CPCR in the presence of model misspecification.

To further investigate the estimator mis-specification and truncation bias, we vary $rank(\hat{\boldsymbol{U}})$ from 1 to 20 while keeping $rank(\boldsymbol{U}) = 20$ and plot the risk of the three algorithms in Figure 7.

Results from Figure 7 show that $\text{rank}(\hat{\boldsymbol{U}})$ does not significantly affect the empirical risk of CPCR if κ is large enough, again confirming the robustness of CPCR against misspscification.

5.2 Kernel Regression with Nystrom Features

Datasets and Baselines. We evaluate our method on 5 benchmark datasets from the UCI Machine Learning Repository (Asuncion et al., 2007). Specifically, we consider Parkinson's disease detection (D1), energy efficiency (D2), Istanbul stock exchange (D3), concrete slump test (D4), and QSAR aquatic toxicity (D5). To ensure overparameterization, we augment the feature space using the Nyström method (Yang et al., 2012). The details of experimental setup are discussed in the supplementary materials.

We compare CPCR against nine widely used latent factor–based methods, including dimension reduction techniques, regularized regression models, and nonparametric kernel approaches. Specifically, these baselines consist of PCR, ridge regression, Ledoit–Wolf OLS (LW-OLS) (Ledoit and Wolf, 2004) Debiased Lasso (De-Lasso) (Van de Geer et al., 2014), kernel regression (Kernel Reg) (Ullah and Pagan, 1999), kernel ridge regression (Kernel Ridge Reg) (Murphy, 2012), sliced inverse regression (SIR) (Li, 1991), supervised PCA (Bair et al., 2006). For every method, both the number of latent variables (when applicable) and the regularization parameters are selected independently using 5-fold cross-validation over each method's own hyperparameter grid.

Results. Table 1 reports results evaluated by root-mean-square error (RMSE) and coefficient of determination (R^2). The results indicate that CPCR consistently achieves better or competitive performance compared to the benchmark algorithms. The reported results are mean values over 10 repeated runs.

5.3 Classification with Vision Foundation Model

Dataset and methodology. We evaluate CPCR on a high-dimensional classification task using the PACS dataset (Li et al., 2017), which consists of images from four distinct visual styles: Photo (P), Art Painting (A), Cartoon (C), and Sketch (S). Each image belongs to one of seven object categories (dog, elephant, giraffe, guitar, horse, house, person). Since the label depends on semantic content rather than artistic style, it is

Table 1: RMSE and R^2 scores for 9 baselines compared against CPCR. CPCR consistently achieves better or competitive performance compared to the baselines with respect to both metrics. Mean values over 10 repeated runs are reported. `NV` stands for negative values.

		CPCR	PCR	Ridge	Ridgeless	LW-OLS	De-Lasso	Kernel Reg	Kernel Ridge	SIR	Super PCA
D1	RMSE	**0.31**	0.48	**0.32**	0.69	**0.32**	20.94	0.34	0.35	0.73	0.37
	R^2	**0.48**	NV	0.44	NV	0.45	NV	0.26	0.27	NV	0.20
D2	RMSE	**0.16**	0.36	0.26	1.72	2.22	2.18	0.27	0.26	6.99	0.23
	R^2	**0.97**	0.86	0.93	0.96	0.95	NV	0.93	0.93	NV	0.95
D3	RMSE	**0.31**	0.33	0.33	3.64	0.40	330.55	0.54	0.41	1.00	0.35
	R^2	**0.88**	0.86	0.86	NV	0.79	NV	0.69	0.82	NV	0.87
D4	RMSE	**0.20**	0.25	**0.21**	0.59	**0.21**	9.36	0.27	0.27	0.32	0.28
	R^2	**0.52**	0.30	0.49	NV	0.50	NV	0.10	0.12	NV	0.02
D5	RMSE	**0.65**	3.99	0.66	0.96	1.24	170.29	0.73	0.64	4.93	0.65
	R^2	**0.59**	NV	0.57	0.06	0.45	NV	0.37	0.52	NV	0.50

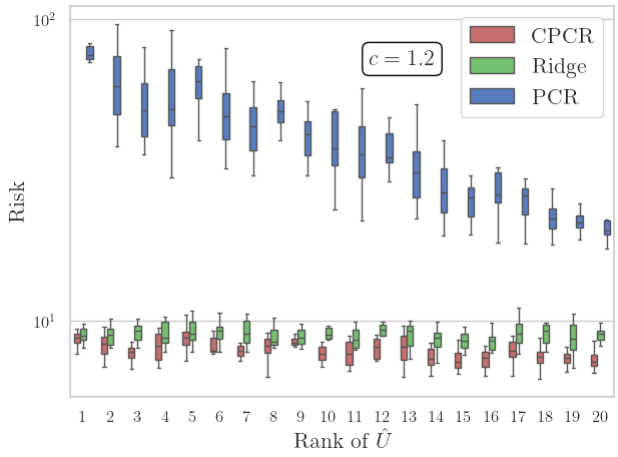

Figure 7: Empirical risks of CPCR, ridge regression and PCR against different rank($\hat{\boldsymbol{U}}$) with $c = 1.2$ and $\kappa = 0.98$. PCR benefits from more PC while CPCR and ridge regression is not sensitive to this parameter.

natural to expect that the predictive signal lies in a low-dimensional subspace of the high-dimensional feature space.

We extract image embeddings using the latest DINOv3 vision foundation models (Siméoni et al., 2025). These models are based on ConvNeXt backbones (Liu et al., 2022) and produce feature vectors with dimensions ranging from 768 to 1536. To reflect the overparameterized regime, we restrict supervision to fewer than 500 labeled examples for training and use the remaining ∼9500 labeled images for evaluation. To simulate noise in the response, we randomly flip 20% of the training labels. The subspace $\hat{\boldsymbol{U}}$ is learned from all available unlabeled features, with dimension fixed at $r = 8$.

Results. Table 2 reports test accuracy across logistic regression (LR), principal component regression (PCR), and our calibrated variant (CPCR). CPCR consistently improves upon both LR and PCR across all feature extractors, despite using only 8 principal components. These results highlight CPCR's ability to mitigate PCR's truncation bias and achieve stable gains in modern overparameterized vision settings.

Table 2: Test accuracy on PACS dataset using ConvNeXt features from DINOv3. CPCR (with $r = 8$ PCs) consistently outperforms LR and PCR across model scales. Mean ± standard deviation over 5 repeated runs is reported.

Model	LR	PCR	CPCR
ConvNext-tiny	90.19 ± 0.40	86.88 ± 2.14	**92.91 ± 1.78**
ConvNext-small	90.90 ± 1.13	86.96 ± 1.30	**93.78 ± 0.76**
ConvNext-base	92.15 ± 0.56	90.33 ± 1.12	**95.28 ± 0.38**
ConvNext-large	93.15 ± 0.67	92.33 ± 0.91	**96.25 ± 0.42**

6 DISCUSSION AND CONCLUSION

We propose CPCR, a principled extension of PCR that leverages cross-fitting to mitigate truncation bias induced by hard spectral cutoff. CPCR enjoys provable guarantees and achieves stable and robust improvements across a range of settings.

A limitation of our current analysis is that it focuses exclusively on in-distribution risk. In many practical scenarios, however, the test distribution may differ from the training distribution, as commonly encountered in domain generalization problems (Gulrajani and Lopez-Paz, 2021; Zhu et al., 2026). Extending the CPCR framework and its theoretical guarantees to settings with distribution shift, particularly under covariate or posterior shift, remains an important direction for future work and may further broaden its applicability.

Acknowledgements

The authors would like to thank Laura Balzano and Clayton Scott for helpful discussions. Yilun Zhu acknowledges the support from the Department of Defense, Defense Threat Reduction Agency under award HDTRA1-20-2-0002.

References

Anish Agarwal, Devavrat Shah, Dennis Shen, and Dogyoon Song. On robustness of principal component regression. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/923e325e16617477e457f6a468a2d6df-Paper.pdf.

Anish Agarwal, Keegan Harris, Justin Whitehouse, and Steven Wu. Adaptive principal component regression with applications to panel data. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=PmqBJ02V1p.

Oguz Akbilgic, Hamparsum Bozdogan, and M Erdal Balaban. A novel hybrid rbf neural networks model as a forecaster. *Statistics and Computing*, 24(3): 365–375, 2014.

Alnur Ali, J. Zico Kolter, and Ryan J. Tibshirani. A continuous-time view of early stopping for least squares, 2019. URL https://arxiv.org/abs/1810.10082.

Arthur Asuncion, David Newman, et al. Uci machine learning repository, 2007.

Francis Bach. High-dimensional analysis of double descent for linear regression with random projections, 2023. URL https://arxiv.org/abs/2303.01372.

Eric Bair, Trevor Hastie, Debashis Paul, and Robert Tibshirani. Prediction by supervised principal components. *Journal of the American Statistical Association*, 101(473):119–137, 2006.

Peter L Bartlett, Philip M Long, Gábor Lugosi, and Alexander Tsigler. Benign overfitting in linear regression. *Proceedings of the National Academy of Sciences*, 117(48):30063–30070, 2020.

Christophe Bonneville, Xiaolong He, April Tran, Jun Sur Park, William Fries, Daniel A Messenger, Siu Wun Cheung, Yeonjong Shin, David M Bortz, Debojyoti Ghosh, et al. A comprehensive review of latent space dynamics identification algorithms for intrusive and non-intrusive reduced-order-modeling. *arXiv preprint arXiv:2403.10748*, 2024.

Betty Jo Casey, Tariq Cannonier, May I Conley, Alexandra O Cohen, Deanna M Barch, Mary M Heitzeg, Mary E Soules, Theresa Teslovich, Danielle V Dellarco, Hugh Garavan, et al. The adolescent brain cognitive development (abcd) study: imaging acquisition across 21 sites. *Developmental cognitive neuroscience*, 32:43–54, 2018.

Matteo Cassotti, Davide Ballabio, Viviana Consonni, Andrea Mauri, Igor V Tetko, and Roberto Todeschini. Prediction of acute aquatic toxicity toward daphnia magna by using the ga-k nn method. *Alternatives to Laboratory Animals*, 42(1):31–41, 2014.

Chen Cheng and Andrea Montanari. Dimension free ridge regression, 2025. URL https://arxiv.org/abs/2210.08571.

Victor Chernozhukov, Denis Chetverikov, Mert Demirer, Esther Duflo, Christian Hansen, Whitney Newey, and James Robins. Double/debiased machine learning for treatment and structural parameters, 2018.

Romain Couillet and Zhenyu Liao. *Random matrix methods for machine learning*. Cambridge University Press, 2022.

K.R. Davidson and Stanislaw Szarek. Local operator theory, random matrices and banach spaces. *Handbook on the Geometry of Banach spaces, Vol. 1*, pages 317–366, 01 2003.

Edgar Dobriban and Stefan Wager. High-dimensional asymptotics of prediction: Ridge regression and classification. *The Annals of Statistics*, 46(1):247 – 279, 2018. doi: 10.1214/17-AOS1549. URL https://doi.org/10.1214/17-AOS1549.

Daniel Gedon, Antonio H. Ribeiro, and Thomas B. Schön. No double descent in principal component regression: A high-dimensional analysis. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp, editors, *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 15271–15293. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/gedon24a.html.

Alden Green and Elad Romanov. The high-dimensional asymptotics of principal component regression. *arXiv preprint arXiv:2405.11676*, 2024.

Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. In *International Conference on Learning Representations*, 2021.

Trevor Hastie, Andrea Montanari, Saharon Rosset, and Ryan J. Tibshirani. Surprises in high-dimensional ridgeless least squares interpolation. *The Annals*

of Statistics, 50(2):949 – 986, 2022. doi: 10.1214/21-AOS2133. URL https://doi.org/10.1214/21-AOS2133.

Arthur E Hoerl and Robert W Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.

Xinting Hu, Yulei Niu, Chunyan Miao, Xian-Sheng Hua, and Hanwang Zhang. On non-random missing labels in semi-supervised learning, 2022. URL https://arxiv.org/abs/2206.14923.

Wenlong Ji, Lihua Lei, and Tijana Zrnic. Predictions as surrogates: Revisiting surrogate outcomes in the age of ai, 2025. URL https://arxiv.org/abs/2501.09731.

Olivier Ledoit and Michael Wolf. A well-conditioned estimator for large-dimensional covariance matrices. *Journal of multivariate analysis*, 88(2):365–411, 2004.

Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Deeper, broader and artier domain generalization. In *Proceedings of the IEEE international conference on computer vision*, pages 5542–5550, 2017.

Ker-Chau Li. Sliced inverse regression for dimension reduction. *Journal of the American Statistical Association*, 86(414):316–327, 1991.

Lexin Li and Xiangrong Yin. Sliced inverse regression with regularizations. *Biometrics*, 64(1):124–131, 2008.

Max Little, Patrick McSharry, Eric Hunter, Jennifer Spielman, and Lorraine Ramig. Suitability of dysphonia measurements for telemonitoring of parkinson's disease. *Nature Precedings*, pages 1–1, 2008.

Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11976–11986, 2022.

Romit Maulik, Themistoklis Botsas, Nesar Ramachandra, Lachlan R Mason, and Indranil Pan. Latent-space time evolution of non-intrusive reduced-order models using gaussian process emulation. *Physica D: Nonlinear Phenomena*, 416:132797, 2021.

Andrew D McRae, Santhosh Karnik, Mark Davenport, and Vidya K Muthukumar. Harmless interpolation in regression and classification with structured features. In *International Conference on Artificial Intelligence and Statistics*, pages 5853–5875. PMLR, 2022.

Marko Medvedev, Gal Vardi, and Nathan Srebro. Overfitting behaviour of gaussian kernel ridgeless regression: Varying bandwidth or dimensionality. *Advances in Neural Information Processing Systems*, 37:52624–52669, 2024.

Kevin P Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021.

Stefano Recanatesi, Matthew Farrell, Guillaume Lajoie, Sophie Deneve, Mattia Rigotti, and Eric Shea-Brown. Predictive learning as a network mechanism for extracting low-dimensional latent space representations. *Nature communications*, 12(1):1417, 2021.

Alexander Ritchie, Laura Balzano, Daniel Kessler, Chandra S Sripada, and Clayton Scott. Supervised PCA: A multiobjective approach. *arXiv preprint arXiv:2011.05309*, 2020.

Hugo Schmutz, Olivier Humbert, and Pierre-Alexandre Mattei. Don't fear the unlabelled: safe semi-supervised learning via simple debiasing, 2023. URL https://arxiv.org/abs/2203.07512.

Jack W Silverstein and Zhi Dong Bai. On the empirical distribution of eigenvalues of a class of large dimensional random matrices. *Journal of Multivariate analysis*, 54(2):175–192, 1995.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. Dinov3, 2025. URL https://arxiv.org/abs/2508.10104.

Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 58(1):267–288, 1996.

Ryan Tibshirani. High-dimensional regression: Ridge. https://www.stat.berkeley.edu/~ryantibs/statlearn-s24/lectures/ridge.pdf, 2024. Lecture notes, Advanced Topics in Statistical Learning, University of California, Berkeley.

Athanasios Tsanas and Angeliki Xifara. Accurate quantitative estimation of energy performance of residential buildings using statistical machine learning tools. *Energy and buildings*, 49:560–567, 2012.

Alexander Tsigler and Peter L Bartlett. Benign overfitting in ridge regression. *Journal of Machine Learning Research*, 24(123):1–76, 2023.

Aman Ullah and Adrian Pagan. *Nonparametric econometrics*. Cambridge university press Cambridge, 1999.

Sara Van de Geer, Peter Bühlmann, Ya'acov Ritov, and Ruben Dezeure. On asymptotically optimal confidence regions and tests for high-dimensional models. *The Annals of Statistics*, 2014.

Yan Wu and Kun Zhang. Tools for the analysis of high-dimensional single-cell rna sequencing data. *Nature Reviews Nephrology*, 16(7):408–421, 2020.

Tianbao Yang, Yu-Feng Li, Mehrdad Mahdavi, Rong Jin, and Zhi-Hua Zhou. Nyström method vs random fourier features: A theoretical and empirical comparison. *Advances in neural information processing systems*, 2012.

I-Cheng Yeh. Concrete Slump Test. UCI Machine Learning Repository, 2007. DOI: https://doi.org/10.24432/C5FG7D.

Yufei Yi and Matey Neykov. A new perspective on debiasing linear regressions. *arXiv preprint arXiv:2104.03464*, 2021.

Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*, 2022.

Lu Yuan, Dongdong Chen, Yi-Ling Chen, Noel Codella, Xiyang Dai, Jianfeng Gao, Houdong Hu, Xuedong Huang, Boxin Li, Chunyuan Li, et al. Florence: A new foundation model for computer vision. *arXiv preprint arXiv:2111.11432*, 2021.

Yuyang Zhang, Shahriar Talebi, and Na Li. Learning low-dimensional latent dynamics from high-dimensional observations: Non-asymptotics and lower bounds. In *International Conference on Machine Learning*, pages 59851–59896. PMLR, 2024.

Yilun Zhu, Jianxin Zhang, Aditya Gangrade, and Clay Scott. Label noise: Ignorance is bliss. *Advances in Neural Information Processing Systems*, 38:116575–116616, 2024.

Yilun Zhu, Naihao Deng, Naichen Shi, Aditya Gangrade, and Clayton Scott. Domain generalization under posterior drift, 2026. URL `https://arxiv.org/abs/2510.04441`.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model.
   Yes
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm.
   No
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes/No/Not Applicable]

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results.
   Yes
   (b) Complete proofs of all theoretical results.
   Yes, in the supplementary materials.
   (c) Clear explanations of any assumptions.
   Yes

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL).
   Yes, in the supplementary materials.
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen).
   Yes
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times).
   Yes
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider).
   Yes, in the supplementary materials.

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets.
   Not Applicable
   (b) The license information of the assets, if applicable.
   Not Applicable
   (c) New assets either in the supplemental material or as a URL, if applicable.
   Not Applicable
   (d) Information about consent from data providers/curators.
   Not Applicable

(e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content.
Not Applicable

5. If you used crowdsourcing or conducted research with human subjects, check if you include:

(a) The full text of instructions given to participants and screenshots.
Not Applicable

(b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable.
Not Applicable

(c) The estimated hourly wage paid to participants and the total amount spent on participant compensation.
Not Applicable

Calibrated Principal Component Regressions: Supplementary Materials

This supplementary material is organized as follows. Section A restates key results from random matrix theory that are used throughout the proofs. Section B presents two auxiliary lemmas that support the main theoretical results, whose proofs are given in Section C. Finally, Section D provides additional experimental details.

A RANDOM MATRIX THEORY

Before delve into our proofs, we remind several related definitions and useful theorems from the random matrix theory in this section. The first very important tool is the Stieltjes transform, where we define as follow:

Definition 2 (Stieltjes transform.) *For a real probability measure μ with support $supp(\mu)$. For all $z \in \mathbb{C} \setminus supp(\mu)$ the Stieltjes transform $m_\mu(z)$ is defined as*

$$m_\mu(z) \equiv \int \frac{1}{t-z}\mu(dt)\,.$$

Knowing Stieltjes transform allows us to state one of the most important theorems in random matrix theory, which is the Marchenko–Pastur theorem. Here, we will introduce a slightly varied version, which is the generalized Marchenko–Pastur theorem.

Notations. In the following theorem and remaining parts of this supplementary material, we use the superscript p to indicate the finite-p companion Stieltjes transform $\tilde{m}^p(z)$, and the subscript z to indicate the derivative with respect to z, $\partial_z \tilde{m}(z) = \tilde{m}_z(z)$. $\tilde{m}(z)$ is the limit of $\tilde{m}^p(z)$ such that $n, p \to \infty$ with $p/n \to c \in (0, \infty)$. We denote $X \leftrightarrow Y$ if, for all bounded norm $\boldsymbol{A} \in \mathbb{R}^{n\times n}$, and $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^n$, $\frac{1}{n}\operatorname{tr} \boldsymbol{A}(\boldsymbol{X}-\boldsymbol{Y}) \xrightarrow{a.s.} 0$, $\boldsymbol{a}^\top(\boldsymbol{X}-\boldsymbol{Y})\boldsymbol{b} \xrightarrow{a.s.} 0$, and $\|\mathbb{E}[\boldsymbol{X}-\boldsymbol{Y}]\| \to 0$. Finally, we denote $\boldsymbol{X}_n \asymp \boldsymbol{Y}_n$ of matrices with growing dimensions if they are equivalent, that is $\lim_{n\to\infty} \frac{1}{n}\operatorname{tr} \boldsymbol{A}_n(\boldsymbol{X}_n - \boldsymbol{Y}_n) \xrightarrow{a.s.} 0$ for any sequence of $\boldsymbol{A}_n$ with bounded norms.

Theorem 3 (Generalized Marchenko–Pastur theorem, Silverstein and Bai (1995)) *Let $\boldsymbol{X} = \boldsymbol{\Sigma}^{\frac{1}{2}}\boldsymbol{Z} \in \mathbb{R}^{p\times n}$, where entries of $\boldsymbol{Z}$ have independent zero mean and unit variance, and light tail. $\boldsymbol{\Sigma}\mathbb{R}^{p\times p}$ is a symmetric nonnegative definite population covariance matrix. Assume that as $n, p \to \infty$ with $p/n \to c \in (0, \infty)$. Letting $\boldsymbol{Q}(z) = (\frac{1}{n}\boldsymbol{X}\boldsymbol{X}^\top - z\boldsymbol{I}_p)^{-1}$ and $\tilde{\boldsymbol{Q}}(z)(\frac{1}{n}\boldsymbol{X}^\top\boldsymbol{X} - z\boldsymbol{I}_n)^{-1}$, we have*

$$\begin{aligned} \boldsymbol{Q}(z) \leftrightarrow \bar{\boldsymbol{Q}}(z) &= -\frac{1}{z}(\boldsymbol{I}_p + \tilde{m}^p(z)\boldsymbol{\Sigma})^{-1}, \\ \tilde{\boldsymbol{Q}}(z) \leftrightarrow \bar{\tilde{\boldsymbol{Q}}}(z) &= \tilde{m}^p(z)\boldsymbol{I}_n\,, \end{aligned} \tag{13}$$

where $(z, \tilde{m}_p(z))$ is the unique solution in $\mathcal{Z}(\mathbb{C}\setminus\mathbb{R}^+)$ of

$$\tilde{m}^p(z) = \left(-z + \frac{1}{n}\operatorname{tr} \boldsymbol{\Sigma}(\boldsymbol{I} + \tilde{m}^p(z)\boldsymbol{\Sigma})^{-1}\right)^{-1}. \tag{14}$$

In particular, if the empirical spectral measure of $\boldsymbol{\Sigma}$ converges, i.e., $\mu_{\boldsymbol{\Sigma}} \to \nu$ as $p \to \infty$, then $\mu_{\frac{1}{n}\boldsymbol{X}\boldsymbol{X}^\top} \xrightarrow{a.s.} \mu$, $\mu_{\frac{1}{n}\boldsymbol{X}^\top\boldsymbol{X}} \xrightarrow{a.s.} \tilde{\mu}$ as $n, p \to \infty$, where μ and $\tilde{\mu}$ are the unique measure having Stieltjes transforms $m(z)$ and $\tilde{m}(z)$, respectively, with the forms

$$\begin{aligned} m(z) &= \frac{1}{c}\tilde{m}(z) + \frac{1-c}{cz}\,, \\ \tilde{m}(z) &= \left(-z + c\int \frac{t\nu(dt)}{1+\tilde{m}(z)t}\right)^{-1}. \end{aligned} \tag{15}$$

B ADDITIONAL LEMMAS

In this section, we introduce two auxiliary lemmas to support the proof of Theorem 4. These two technical lemmas allow us to deal with the correlation between $\hat{\gamma}_1^{\texttt{calib}}$ and $\hat{\gamma}_2^{\texttt{calib}}$.

Lemma 1 (Vanishing direction of estimator.) *Under the same settings as Theorem 4, denote* $\hat{\boldsymbol{\Sigma}}_1 = \frac{1}{n_1}\boldsymbol{X}_1\boldsymbol{X}_1^\top$ *the sample covariance of one random split of* $\boldsymbol{X}$*. Then,*

$$\boldsymbol{U}(\boldsymbol{U}^\top\hat{\boldsymbol{\Sigma}}_1\boldsymbol{U})^{-1}\boldsymbol{U}^\top\hat{\boldsymbol{\Sigma}}_1(\boldsymbol{\Pi}_{\boldsymbol{U}}-\boldsymbol{I})\boldsymbol{\gamma}^* \xrightarrow{a.s.} \mathbf{0}\,.$$

Proof. Recall from Couillet and Liao (2022, Ch. 2, Sec. 2.4.2, p.109)

$$\boldsymbol{a}^\top f\left(\frac{1}{n}\boldsymbol{X}\boldsymbol{X}^T\right)\boldsymbol{b} = \frac{1}{2\pi i}\oint_\Gamma \frac{f(z)}{z}\boldsymbol{a}^\top(\boldsymbol{I}+\tilde{m}(z)\boldsymbol{\Sigma})^{-1}\boldsymbol{b}\,dz + o(1)\,,$$

where $f:\mathbb{C}\to\mathbb{C}$ analytic on the neighborhood of eigenvalue of $\frac{1}{n}\boldsymbol{X}\boldsymbol{X}^T$, Γ is a contour circling around the limiting spectral support $\mathrm{supp}(\mu)$ (μ as in Theorem 3), and $\boldsymbol{\Sigma}$ is the population covariance of $\boldsymbol{X}$. Here, if we take $\boldsymbol{a}=\boldsymbol{U}$, $\boldsymbol{b}=\boldsymbol{I}-\boldsymbol{\Pi}_{\boldsymbol{U}}$, and $f(z)=z$, we have

$$\begin{aligned}\boldsymbol{U}^\top\hat{\boldsymbol{\Sigma}}_1(\boldsymbol{I}-\boldsymbol{\Pi}_{\boldsymbol{U}}) =& \frac{1}{2\pi i}\oint_\Gamma \boldsymbol{U}^\top(\boldsymbol{I}+\tilde{m}(z)\boldsymbol{\Sigma})^{-1}(\boldsymbol{I}-\boldsymbol{\Pi}_{\boldsymbol{U}})\,dz + o(1)\\ =& \frac{1}{2\pi i}\oint_\Gamma \boldsymbol{U}^\top(\boldsymbol{U}(I+\tilde{m}(z)\boldsymbol{\Sigma}_s)^{-1}\boldsymbol{U}^\top + \boldsymbol{V}(I+\tilde{m}(z)\boldsymbol{\Sigma}_c)^{-1}\boldsymbol{V}^\top)(\boldsymbol{I}-\boldsymbol{\Pi}_{\boldsymbol{U}})\,dz + o(1).\end{aligned}$$

Now, because $\boldsymbol{U}^\top(\boldsymbol{I}-\boldsymbol{\Pi}_{\boldsymbol{U}})=\mathbf{0}$ and $\boldsymbol{U}^\top\boldsymbol{V}=\mathbf{0}$, the integrand is $\mathbf{0}$. Therefore,

$$\hat{\boldsymbol{U}}(\hat{\boldsymbol{U}}^\top\hat{\boldsymbol{\Sigma}}_1\hat{\boldsymbol{U}})^{-1}\hat{\boldsymbol{U}}^\top\hat{\boldsymbol{\Sigma}}_1(\boldsymbol{\Pi}_{\boldsymbol{U}}-\boldsymbol{I})\boldsymbol{\gamma}^* \xrightarrow{a.s.} \mathbf{0}\,.$$

■

Lemma 2 (Vanishing variance.) *Under the same settings as Theorem 4, denote* $\hat{\boldsymbol{\Sigma}}_2 = \frac{1}{n_2}\boldsymbol{X}_2\boldsymbol{X}_2^\top$ *the sample covariance of one random split of* $\boldsymbol{X}$*. Then, with high probability,*

$$\frac{\lambda^2\sigma^2}{n_2}\,\mathrm{tr}((\hat{\boldsymbol{\Sigma}}_2+\lambda\boldsymbol{I})^{-1}\Sigma(\hat{\boldsymbol{\Sigma}}_2+\lambda\boldsymbol{I})^{-1}\boldsymbol{U}(\boldsymbol{U}^\top\hat{\boldsymbol{\Sigma}}_1\boldsymbol{U})^{-1}\boldsymbol{U}^\top) \le \mathcal{O}(\frac{r}{n})$$

Proof. We will tackle by bounding $(\boldsymbol{U}^\top\hat{\boldsymbol{\Sigma}}_1\boldsymbol{U})^{-1}$ and $\boldsymbol{U}^\top(\hat{\boldsymbol{\Sigma}}_2+\lambda\boldsymbol{I})^{-1}\Sigma(\hat{\boldsymbol{\Sigma}}_2+\lambda\boldsymbol{I})^{-1}\boldsymbol{U}$.

Write $\boldsymbol{U}^\top\hat{\boldsymbol{\Sigma}}_1\boldsymbol{U} = \frac{1}{n_1}\boldsymbol{U}^\top\boldsymbol{X}_1\boldsymbol{X}_1^\top\boldsymbol{U}$. Then $\boldsymbol{\Sigma}_{\boldsymbol{U}} = \mathbb{E}(\frac{1}{n_1}\boldsymbol{U}^\top\boldsymbol{X}_1\boldsymbol{X}_1^\top\boldsymbol{U}) = \boldsymbol{U}^\top\boldsymbol{\Sigma}\boldsymbol{U} = \boldsymbol{\Sigma}_s$ (Because $\mathbb{E}(\hat{\boldsymbol{\Sigma}}_1) = \frac{1}{n_1}\sum_{i=1}^{n_1}\mathbb{E}(\boldsymbol{x}_i\boldsymbol{x}_i^\top) = \mathbb{E}(\boldsymbol{x}_1\boldsymbol{x}_1^\top) = \boldsymbol{\Sigma}$). Define $\boldsymbol{W} = \boldsymbol{\Sigma}_{\boldsymbol{U}}^{-1/2}\boldsymbol{U}^\top\boldsymbol{X}_1$, and write $\frac{1}{n_1}\boldsymbol{U}^\top\boldsymbol{X}_1\boldsymbol{X}_1^\top\boldsymbol{U} = \Sigma_{\boldsymbol{U}}^{1/2}\frac{1}{n_1}\boldsymbol{W}\boldsymbol{W}^\top\boldsymbol{\Sigma}_{\boldsymbol{U}}^{1/2}$. Therefore,

$$\begin{aligned}\left\|(\boldsymbol{U}^\top\hat{\boldsymbol{\Sigma}}_1\boldsymbol{U})^{-1}\right\| = \left\|(\boldsymbol{\Sigma}_{\boldsymbol{U}}^{1/2}\frac{1}{n_1}\boldsymbol{W}\boldsymbol{W}^\top\boldsymbol{\Sigma}_{\boldsymbol{U}}^{1/2})^{-1}\right\| &\le \left\|\boldsymbol{\Sigma}_{\boldsymbol{U}}^{-1}\right\|\left\|\left(\frac{1}{n_1}\boldsymbol{W}\boldsymbol{W}^\top\right)^{-1}\right\|\\ &= \frac{1}{\lambda_{\min}(\boldsymbol{\Sigma}_{\boldsymbol{U}})}\frac{1}{\lambda_{\min}\left(\frac{1}{n_1}\boldsymbol{W}\boldsymbol{W}^\top\right)}\end{aligned}$$

by the multidisciplinary of matrix norm. Note that $\lambda_{\min}(\boldsymbol{\Sigma}_{\boldsymbol{U}}) = \lambda_{\min}(\boldsymbol{\Sigma}_s)$. Also note that each column $\boldsymbol{w}$ of $\boldsymbol{W} = \boldsymbol{\Sigma}_{\boldsymbol{U}}^{-1/2}\boldsymbol{U}^\top\boldsymbol{H}^{1/2}\boldsymbol{Z}_1$ has covariance $\mathbb{E}(\boldsymbol{w}\boldsymbol{w}^\top) = \boldsymbol{\Sigma}_{\boldsymbol{U}}^{-1/2}\boldsymbol{U}^\top\boldsymbol{H}\boldsymbol{U}\boldsymbol{\Sigma}_{\boldsymbol{U}}^{-1/2} = \boldsymbol{I}$ and mean $\mathbf{0}$. From Davidson and Szarek (2003) Theorem II.13,

$$\mathbb{P}\left(\lambda_{\min}\left(\frac{1}{n_1}\boldsymbol{W}\boldsymbol{W}^\top\right) > (1-\sqrt{r/n_1}-t)^2\right) \ge 1-\exp\left(\frac{-n_1t^2}{2}\right)\,.$$

So, with high probability (choose $t = n_1^{-1/4}$),

$$\left\|(\boldsymbol{U}^\top \hat{\boldsymbol{\Sigma}}_1 \boldsymbol{U})^{-1}\right\| \leq \frac{1}{\lambda_{\min}(\boldsymbol{\Sigma}_s)} \frac{1}{(1-\sqrt{r/n_1}-t)^2}\,.$$

Next, we bound $\boldsymbol{U}^\top(\hat{\boldsymbol{\Sigma}}_2+\lambda \boldsymbol{I})^{-1}\Sigma(\hat{\boldsymbol{\Sigma}}_2+\lambda \boldsymbol{I})^{-1}\boldsymbol{U}$:

$$\begin{aligned}\left\|\boldsymbol{U}^\top(\hat{\boldsymbol{\Sigma}}_2+\lambda \boldsymbol{I})^{-1}\Sigma(\hat{\boldsymbol{\Sigma}}_2+\lambda \boldsymbol{I})^{-1}\boldsymbol{U}\right\| &\leq \left\|(\hat{\boldsymbol{\Sigma}}_2+\lambda \boldsymbol{I})^{-1}\right\|^2 \|\boldsymbol{\Sigma}\| \\ &= \frac{\|\boldsymbol{\Sigma}\|}{\lambda_{\min}(\hat{\boldsymbol{\Sigma}}_2+\lambda \boldsymbol{I})^2} \leq \frac{\|\boldsymbol{\Sigma}\|}{\lambda^2}\end{aligned}$$

Put together,

$$\begin{aligned}\operatorname{tr}((\hat{\boldsymbol{\Sigma}}_2+\lambda \boldsymbol{I})^{-1}\Sigma(\hat{\boldsymbol{\Sigma}}_2+\lambda \boldsymbol{I})^{-1}\boldsymbol{U}(\boldsymbol{U}^\top\hat{\boldsymbol{\Sigma}}_1\boldsymbol{U})^{-1}\boldsymbol{U}^\top) &\leq |\operatorname{tr}(\boldsymbol{U}^\top(\hat{\boldsymbol{\Sigma}}_2+\lambda \boldsymbol{I})^{-1}\boldsymbol{\Sigma}(\hat{\boldsymbol{\Sigma}}_2+\lambda \boldsymbol{I})^{-1}\boldsymbol{U})| \left\|(\boldsymbol{U}^\top\hat{\boldsymbol{\Sigma}}_1\boldsymbol{U})^{-1}\right\| \\ &\leq r \cdot \left\|\boldsymbol{U}^\top(\hat{\boldsymbol{\Sigma}}_2+\lambda \boldsymbol{I})^{-1}\boldsymbol{\Sigma}(\hat{\boldsymbol{\Sigma}}_2+\lambda \boldsymbol{I})^{-1}\boldsymbol{U}\right\| \left\|(\boldsymbol{U}^\top\hat{\boldsymbol{\Sigma}}_1\boldsymbol{U})^{-1}\right\| \\ &\leq r \cdot \frac{\|\boldsymbol{\Sigma}\|}{\lambda^2}\frac{1}{\lambda_{\min}(\boldsymbol{\Sigma}_s)}\frac{1}{(1-\sqrt{r/n_1}-t)^2}\,,\end{aligned}$$

where the first inequality can be obtained either from Holder's inequality or von Neumann's trace inequality. The second inequality is from $|\operatorname{tr}(\boldsymbol{A})| = |\sum_i \lambda_i(\boldsymbol{A})| \leq r \cdot |\lambda_{\max}(\boldsymbol{A})| = r \cdot \|\boldsymbol{A}\|_{op}$ for $\boldsymbol{A}$ PSD.

Note that $n_1 = n_2 = \frac{n}{2}$. Therefore,

$$\begin{aligned}\frac{\lambda^2\sigma^2}{n_2}\operatorname{tr}((\hat{\boldsymbol{\Sigma}}_2+\lambda \boldsymbol{I})^{-1}\Sigma(\hat{\boldsymbol{\Sigma}}_2+\lambda \boldsymbol{I})^{-1}\boldsymbol{U}(\boldsymbol{U}^\top\hat{\boldsymbol{\Sigma}}_1\boldsymbol{U})^{-1}\boldsymbol{U}^\top) &\leq \frac{r}{n}\cdot\frac{\|\boldsymbol{\Sigma}\|}{\lambda_{\min}(\boldsymbol{\Sigma}_s)}\frac{2\sigma^2}{(1-\sqrt{r/n_1}-t)^2} \\ &= \mathcal{O}\left(\frac{r}{n}\right)\,.\end{aligned}$$

■

C MAIN THEOREMS

We will first restate our main theorems and prove them in the subsequent subsections. For the following calculations, recall that our out-of-sample prediction risk is defined as:

$$\mathcal{R}(\hat{\boldsymbol{\gamma}}) \triangleq \mathbb{E}_{\boldsymbol{x}_0,\boldsymbol{\gamma}^*}[(\boldsymbol{x}_0^\top \hat{\boldsymbol{\gamma}} - \boldsymbol{x}_0^\top \boldsymbol{\gamma}^*)^2]\,. \tag{16}$$

Theorem 4 (Exact calibration estimator risk.) *In the asymptotic limit where $n, p \to \infty$ such that $p/n \to c \in (1, \infty)$. With high probability, the risk of our calibration estimator $\hat{\boldsymbol{\gamma}}^{\texttt{calib}}$ converge to a deterministic limit:*

$$\mathcal{R}(\hat{\boldsymbol{\gamma}}^{\texttt{CPCR}}) \to B_{\boldsymbol{X}}(\hat{\boldsymbol{\gamma}}^{\texttt{CPCR}}) + V(\hat{\boldsymbol{\gamma}}^{\texttt{CPCR}})\,, \tag{17}$$

where the bias term can be further estimated by

$$\begin{aligned} B_{\boldsymbol{X}}(\hat{\boldsymbol{\gamma}}^{\texttt{CPCR}}) =& \frac{1}{4}(B_{\boldsymbol{X}}(\hat{\boldsymbol{\gamma}}_1^{\texttt{calib}}) + B_{\boldsymbol{X}}(\hat{\boldsymbol{\gamma}}_2^{\texttt{calib}})) \\ &+ \frac{1}{2} B_{\boldsymbol{X}}(\hat{\boldsymbol{\gamma}}_1^{\texttt{calib}}, \hat{\boldsymbol{\gamma}}_2^{\texttt{calib}}) \end{aligned}$$

and the variance term can be estimated by

$$V(\hat{\boldsymbol{\gamma}}^{\texttt{CPCR}}) = \frac{1}{4}(V_{\epsilon_2}(\hat{\boldsymbol{\gamma}}_1^{\texttt{calib}}) + V_{\epsilon_1}(\hat{\boldsymbol{\gamma}}_2^{\texttt{calib}}))\,.$$

Moreover, consider $\{\mu_j(\boldsymbol{\Sigma}_c)\}_{j=1}^{p-r}$ the diagonal entries of $\boldsymbol{\Sigma}_c$ and $\nu_c = \frac{1}{p-r}\sum_{j=1}^{p-r}\delta_{\mu_j(\boldsymbol{\Sigma}_c)}$, then the almost sure limits of each individual terms are

$$\begin{aligned} & B_{\boldsymbol{X}}(\hat{\boldsymbol{\gamma}}_1^{\texttt{calib}}) \xrightarrow{a.s.} \\ &+(1-\kappa)(p-r)\int \frac{\mu}{1+\tilde{m}(-\lambda)\mu} d\nu_c(\mu) \\ &+(1-\kappa)(p-r)\tilde{m}_\rho(-\lambda)\int \frac{\mu}{(1+\tilde{m}(-\lambda)\mu)^2} d\nu_c(\mu) \\ &-(1-\kappa)(p-r)\tilde{m}(-\lambda)\int \frac{\mu^2}{(1+\tilde{m}(-\lambda)\mu)^2} d\nu_c(\mu)\,, \end{aligned} \tag{18}$$

$$\begin{aligned} & B_{\boldsymbol{X}}(\hat{\boldsymbol{\gamma}}_1^{\texttt{calib}}, \hat{\boldsymbol{\gamma}}_2^{\texttt{calib}}) \xrightarrow{a.s.} \\ & (1-\kappa)(p-r)\int \frac{\mu}{(1+\tilde{m}(-\lambda)\mu)^2} d\nu_c(\mu) \end{aligned} \tag{19}$$

and

$$V_{\epsilon_2}(\hat{\boldsymbol{\gamma}}_1^{\texttt{calib}}) \xrightarrow{a.s.} \sigma^2\left(\frac{\tilde{m}_z(-\lambda)}{\tilde{m}^2(-\lambda)} - 1\right). \tag{20}$$

$\tilde{m}(\cdot)$ is defined implicitly as the solutions to a nonlinear equation depending on n, p and $\boldsymbol{\Sigma}$. $\tilde{m}_\rho(\cdot)$ and $\tilde{m}_z(\cdot)$ are partial derivatives with respect to ρ and z respectively.

Theorem 5 (Optimal λ.) *Under the same conditions as Theorem 4, the optimal lambda λ^* to achieve the minimal risk in* (16) *satisfies the following equation:*

$$B'_{\boldsymbol{X}}(\lambda^*) + V'(\lambda^*) = 0\,.$$

In particular,

$$\begin{aligned}B'_{\boldsymbol{X}}(\lambda) =& \frac{(1-\kappa)(p-r)}{2}\bigg(2\int \frac{\mu^2\tilde{m}_z(-\lambda)}{(1+\tilde{m}(-\lambda)\mu)^2}d\nu_c(\mu)\\ &-\int \frac{\mu\tilde{m}_{\rho z}(-\lambda)}{(1+\tilde{m}(-\lambda)\mu)^2}d\nu_c(\mu)\\ &+2\int \frac{\mu^2(\tilde{m}_\rho(-\lambda)\tilde{m}_z(-\lambda)+\tilde{m}_z(-\lambda))}{(1+\tilde{m}(-\lambda)\mu)^3}d\nu_c(\mu)\\ &-2\int \frac{\mu^3\tilde{m}(-\lambda)\tilde{m}_z(-\lambda)}{(1+\tilde{m}(-\lambda)\mu)^3}d\nu_c(\mu)\bigg)\end{aligned}$$

and

$$V'(\lambda) = \frac{\sigma^2}{2}\left(\frac{-\tilde{m}(-\lambda)\tilde{m}_{zz}(-\lambda)+2\tilde{m}_z^2(-\lambda)}{\tilde{m}^3(-\lambda)}\right).$$

$\tilde{m}_{\rho z}(\cdot)$ denotes $\partial_{\rho z}\tilde{m}^p(\cdot)$ and $\tilde{m}_{zz}(\cdot)$ denotes $\partial_{zz}\tilde{m}^p(\cdot)$.

C.1 Proof of Theorem 1

We will break the calculations into three calculations. First, we will give the deterministic equivalents for bias and variance of $\hat{\boldsymbol{\gamma}}_1^{\texttt{calib}}$. Then we will give the deterministic equivalent for $B_{\boldsymbol{X}}(\hat{\boldsymbol{\gamma}}_1^{\texttt{calib}}, \hat{\boldsymbol{\gamma}}_2^{\texttt{calib}})$.

First, fitting a subspace regression model

$$\hat{\xi} = \arg\min_{\xi} \left\|\boldsymbol{y}_1^\top - \xi^\top \hat{\boldsymbol{U}}^\top \boldsymbol{X}_1\right\|_2^2$$

on $(\boldsymbol{X}_1, \boldsymbol{y}_1)$ gives

$$\hat{\xi_1} = \left(\hat{\boldsymbol{U}}^\top \frac{\boldsymbol{X}_1\boldsymbol{X}_1^\top}{n_1}\hat{\boldsymbol{U}}\right)^{-1}\hat{\boldsymbol{U}}^\top\frac{\boldsymbol{X}_1\boldsymbol{y}_1}{n_1}. \tag{21}$$

Then, we calibrate $\boldsymbol{\gamma}$ with $\boldsymbol{\gamma}_1^{\texttt{init}} = \hat{\boldsymbol{U}}\hat{\xi_1}$ and fit

$$\hat{\boldsymbol{\gamma}}_1^{\texttt{calib}} = \arg\min_{\boldsymbol{\gamma}} \left\|\boldsymbol{y}_2^\top - \boldsymbol{\gamma}^\top\boldsymbol{X}_2\right\|_2^2 + \lambda\left\|\boldsymbol{\gamma} - \boldsymbol{\gamma}_1^{\texttt{init}}\right\|_2^2.$$

Substituting $\boldsymbol{y}_1 = \boldsymbol{X}_1^\top\boldsymbol{\gamma}^* + \boldsymbol{\epsilon}_1$ and $\boldsymbol{y}_2 = \boldsymbol{X}_2^\top\boldsymbol{\gamma}^* + \boldsymbol{\epsilon}_2$, we have

$$\begin{aligned}\hat{\boldsymbol{\gamma}}_1^{\texttt{calib}} - \boldsymbol{\gamma}^* &= \left(\left(\frac{\boldsymbol{X}_2\boldsymbol{X}_2^\top}{n_2}+\frac{\lambda}{n_2}\boldsymbol{I}\right)^{-1}\frac{\boldsymbol{X}_2\boldsymbol{X}_2^\top}{n_2} - \boldsymbol{I}\right)\boldsymbol{\gamma}^* + \left(\frac{\boldsymbol{X}_2\boldsymbol{X}_2^\top}{n_2}+\frac{\lambda}{n_2}\boldsymbol{I}\right)^{-1}\left(\frac{\boldsymbol{X}_2\boldsymbol{\epsilon}_2}{n_2}+\frac{\lambda}{n_2}\boldsymbol{\gamma}_1^{\texttt{init}}\right)\\ &= \frac{\lambda}{n_2}(\hat{\boldsymbol{\Sigma}}_2+\frac{\lambda}{n_2}\boldsymbol{I})^{-1}(\boldsymbol{\gamma}_1^{\texttt{init}}-\boldsymbol{\gamma}^*) + (\hat{\boldsymbol{\Sigma}}_2+\frac{\lambda}{n_2}\boldsymbol{I})^{-1}\frac{\boldsymbol{X}_2\boldsymbol{\epsilon}_2}{n_2},\end{aligned} \tag{22}$$

where $\hat{\Sigma}_i = \frac{\boldsymbol{X}_i\boldsymbol{X}_i^\top}{n_i}$ for $i = 1, 2$. The second equality comes from $\left(\hat{\boldsymbol{\Sigma}}_2+\frac{\lambda}{n_2}\boldsymbol{I}\right)^{-1}\hat{\boldsymbol{\Sigma}}_2 - \boldsymbol{I} = \left(\hat{\boldsymbol{\Sigma}}_2+\frac{\lambda}{n_2}\boldsymbol{I}\right)^{-1}\left(\hat{\boldsymbol{\Sigma}}_2+\frac{\lambda}{n_2}\boldsymbol{I}-\frac{\lambda}{n_2}\boldsymbol{I}\right) - \boldsymbol{I} = -\frac{\lambda}{n_2}\left(\hat{\boldsymbol{\Sigma}}_2+\frac{\lambda}{n_2}\boldsymbol{I}\right)^{-1}$. For notation simplicity, from here we rescale $\frac{\lambda}{n_2} = \frac{\lambda}{n_1}$ to λ.

Next, substituting $\boldsymbol{y}_1$ into (21) gives

$$\hat{\xi_1} = (\hat{\boldsymbol{U}}^\top\hat{\boldsymbol{\Sigma}}_1\hat{\boldsymbol{U}})^{-1}\hat{\boldsymbol{U}}^\top\hat{\boldsymbol{\Sigma}}_1\boldsymbol{\gamma}^* + (\hat{\boldsymbol{U}}^\top\hat{\boldsymbol{\Sigma}}_1\hat{\boldsymbol{U}})^{-1}\hat{\boldsymbol{U}}^\top\frac{\boldsymbol{X}_1\boldsymbol{\epsilon}_1^\top}{n_1},$$

and with $\boldsymbol{\gamma}_1^{\texttt{init}} = \hat{\boldsymbol{U}}\hat{\xi_1}$ we have

$$\boldsymbol{\gamma}_1^{\texttt{init}} - \boldsymbol{\gamma}^* = (\hat{\boldsymbol{U}}(\hat{\boldsymbol{U}}^\top\hat{\boldsymbol{\Sigma}}_1\hat{\boldsymbol{U}})^{-1}\hat{\boldsymbol{U}}^\top\hat{\boldsymbol{\Sigma}}_1 - \boldsymbol{I})\boldsymbol{\gamma}^* + \hat{\boldsymbol{U}}(\hat{\boldsymbol{U}}^\top\hat{\boldsymbol{\Sigma}}_1\hat{\boldsymbol{U}})^{-1}\hat{\boldsymbol{U}}^\top\frac{\boldsymbol{X}_1\boldsymbol{\epsilon}_1^\top}{n_1}. \tag{23}$$

Substituting (23) into (22), we have

$$\begin{aligned}&\hat{\gamma}_1^{\texttt{calib}} - \boldsymbol{\gamma}^* \\ =&\lambda(\hat{\boldsymbol{\Sigma}}_2 + \lambda\boldsymbol{I})^{-1}\left[(\boldsymbol{\Pi}_{\boldsymbol{U}} - \boldsymbol{I})\boldsymbol{\gamma}^* + \hat{\boldsymbol{U}}(\hat{\boldsymbol{U}}^\top\hat{\boldsymbol{\Sigma}}_1\hat{\boldsymbol{U}})^{-1}\hat{\boldsymbol{U}}^\top\frac{\boldsymbol{X}_1\epsilon_1^\top}{n_2} + \hat{\boldsymbol{U}}(\hat{\boldsymbol{U}}^\top\hat{\boldsymbol{\Sigma}}_1\hat{\boldsymbol{U}})^{-1}\hat{\boldsymbol{U}}^\top\hat{\boldsymbol{\Sigma}}_1(\boldsymbol{\Pi}_{\boldsymbol{U}} - \boldsymbol{I})\boldsymbol{\gamma}^*\right] + (\hat{\boldsymbol{\Sigma}}_2 + \lambda\boldsymbol{I})^{-1}\frac{\boldsymbol{X}_2\epsilon_2^\top}{n_2}.\end{aligned} \tag{24}$$

By Lemma 1, the red term converges to 0 almost surely. Therefore, the bias term is

$$\begin{aligned}B_{\boldsymbol{X}}(\hat{\gamma}_1^{\texttt{calib}}) &= \lambda^2\operatorname{tr}(\mathbb{E}((\boldsymbol{\Pi}_{\boldsymbol{U}} - \boldsymbol{I})\boldsymbol{\gamma}^*\boldsymbol{\gamma}^{*\top}(\boldsymbol{\Pi}_{\boldsymbol{U}} - \boldsymbol{I}))\cdot(\hat{\boldsymbol{\Sigma}}_2 + \lambda\boldsymbol{I})^{-1}\Sigma(\hat{\boldsymbol{\Sigma}}_2 + \lambda\boldsymbol{I})^{-1}) \\ &= \lambda^2\operatorname{tr}(\mathbb{E}(\boldsymbol{\gamma}_\perp^*\boldsymbol{\gamma}_\perp^{*\top})\cdot(\hat{\boldsymbol{\Sigma}}_2 + \lambda\boldsymbol{I})^{-1}\Sigma(\hat{\boldsymbol{\Sigma}}_2 + \lambda\boldsymbol{I})^{-1}) \\ &= \lambda^2\operatorname{tr}(\Sigma_{\boldsymbol{\gamma}_\perp^*}\cdot(\hat{\boldsymbol{\Sigma}}_2 + \lambda\boldsymbol{I})^{-1}\Sigma(\hat{\boldsymbol{\Sigma}}_2 + \lambda\boldsymbol{I})^{-1}).\end{aligned} \tag{25}$$

Recall that $\boldsymbol{\Sigma}_{\boldsymbol{\gamma}^*} = \kappa\boldsymbol{\Pi}_{\boldsymbol{U}} + (1-\kappa)\boldsymbol{\Pi}_{\boldsymbol{V}}$, giving $\boldsymbol{\Sigma}_{\boldsymbol{\gamma}_\perp^*} = (\boldsymbol{I} - \boldsymbol{\Pi}_{\boldsymbol{U}})\boldsymbol{\Sigma}_{\boldsymbol{\gamma}^*}(\boldsymbol{I} - \boldsymbol{\Pi}_{\boldsymbol{U}}) = (1-\kappa)\boldsymbol{\Pi}_{\boldsymbol{V}}$.

$$B_{\boldsymbol{X}}(\hat{\gamma}_1^{\texttt{calib}}) = \lambda^2(1-\kappa)\operatorname{tr}(\boldsymbol{\Pi}_{\boldsymbol{V}}\cdot(\hat{\boldsymbol{\Sigma}}_2 + \lambda\boldsymbol{I})^{-1}\boldsymbol{\Sigma}(\hat{\boldsymbol{\Sigma}}_2 + \lambda\boldsymbol{I})^{-1}) \tag{26}$$

Observe from (26) that the appearance of $(\hat{\boldsymbol{\Sigma}}_2 + \lambda\boldsymbol{I})^{-1}\boldsymbol{\Sigma}(\hat{\boldsymbol{\Sigma}}_2 + \lambda\boldsymbol{I})^{-1}$. We cannot directly replace both resolvents $(\hat{\boldsymbol{\Sigma}}_2 + \lambda\boldsymbol{I})^{-1}$ with their deterministic equivalents because they are correlated. As a result, we use some derivative tricks following Tibshirani (2024).

We have:

$$\lambda^2(\hat{\boldsymbol{\Sigma}}_2 + \lambda\boldsymbol{I})^{-1}\boldsymbol{\Sigma}(\hat{\boldsymbol{\Sigma}}_2 + \lambda\boldsymbol{I})^{-1} = -\frac{d}{d\rho}\left\{\lambda(\hat{\boldsymbol{\Sigma}}_2 + \lambda(\boldsymbol{I} + \rho\boldsymbol{\Sigma})^{-1})\right\}\Big|_{\rho=0},$$

where the matrix in the curly bracket can be rewritten as

$$\lambda(\hat{\boldsymbol{\Sigma}}_2 + \lambda(\boldsymbol{I} + \rho\boldsymbol{\Sigma})^{-1}) = (\boldsymbol{I} + \rho\boldsymbol{\Sigma})^{-1/2}\lambda(\hat{\boldsymbol{\Sigma}}_\rho + \lambda\boldsymbol{I})^{-1}(\boldsymbol{I} + \rho\boldsymbol{\Sigma})^{-1/2},$$

where we define $\hat{\boldsymbol{\Sigma}}_\rho = \boldsymbol{\Sigma}_\rho^{1/2}\frac{Z_2Z_2^\top}{n_2}\boldsymbol{\Sigma}_\rho^{1/2}$ and $\boldsymbol{\Sigma}_\rho = (\boldsymbol{I} + \rho\boldsymbol{\Sigma})^{-1/2}\boldsymbol{\Sigma}(\boldsymbol{I} + \rho\boldsymbol{\Sigma})^{-1/2}$. Applying generalized MP law, we have

$$\begin{aligned}&\lambda^2\boldsymbol{\Pi}_{\boldsymbol{V}}(\hat{\boldsymbol{\Sigma}}_2 + \lambda\boldsymbol{I})^{-1}\boldsymbol{\Sigma}(\hat{\boldsymbol{\Sigma}}_2 + \lambda\boldsymbol{I})^{-1} \\ =& -\boldsymbol{\Pi}_{\boldsymbol{V}}\cdot\frac{d}{d\rho}\left\{(\boldsymbol{I} + \rho\boldsymbol{\Sigma})^{-1/2}\lambda(\hat{\boldsymbol{\Sigma}}_\rho + \lambda\boldsymbol{I})^{-1}(\boldsymbol{I} + \rho\boldsymbol{\Sigma})^{-1/2}\right\}\Big|_{\rho=0} \\ =& -\boldsymbol{\Pi}_{\boldsymbol{V}}\cdot\frac{d}{d\rho}\left\{(\boldsymbol{I} + \rho\boldsymbol{\Sigma})^{-1/2}\lambda\boldsymbol{Q}(-\lambda;\rho)(\boldsymbol{I} + \rho\boldsymbol{\Sigma})^{-1/2}\right\}\Big|_{\rho=0} \\ \asymp& -\boldsymbol{\Pi}_{\boldsymbol{V}}\cdot\frac{d}{d\rho}\left\{(\boldsymbol{I} + \rho\boldsymbol{\Sigma})^{-1/2}(\boldsymbol{I} + \tilde{m}^p(-\lambda)\boldsymbol{\Sigma}_\rho)^{-1}(\boldsymbol{I} + \rho\boldsymbol{\Sigma})^{-1/2}\right\}\Big|_{\rho=0}\end{aligned}$$

where $\boldsymbol{Q}(z;\rho)$ is the resolvent for $\hat{\boldsymbol{\Sigma}}_\rho$ with its deterministic equivalence $\bar{\boldsymbol{Q}}(z;\rho)$.

Now we proceed to derivative calculations. For simplicity and future reference, we denote $\tilde{m}_\rho(\cdot)$ and $\tilde{m}_z(\cdot)$ as partial derivatives of $\tilde{m}^p(z;\rho)$ with respect to ρ and z respectively.

$$\begin{aligned}&\frac{d}{d\rho}\left\{(\boldsymbol{I} + \rho\boldsymbol{\Sigma})^{-1/2}(\boldsymbol{I} + \tilde{m}^p(-\lambda;\rho)\boldsymbol{\Sigma}_\rho)^{-1}(\boldsymbol{I} + \rho\boldsymbol{\Sigma})^{-1/2}\right\} \\ =& -\frac{1}{2}(\boldsymbol{I} + \rho\boldsymbol{\Sigma})^{-3/2}\boldsymbol{\Sigma}(\boldsymbol{I} + \tilde{m}^p(-\lambda;\rho)\boldsymbol{\Sigma}_\rho)^{-1}(\boldsymbol{I} + \rho\boldsymbol{\Sigma})^{-1/2} - \frac{1}{2}(\boldsymbol{I} + \rho\boldsymbol{\Sigma})^{-1/2}(\boldsymbol{I} + \tilde{m}^p(-\lambda;\rho)\boldsymbol{\Sigma}_\rho)^{-1}(\boldsymbol{I} + \rho\boldsymbol{\Sigma})^{-3/2}\boldsymbol{\Sigma} \\ &-(\boldsymbol{I} + \rho\boldsymbol{\Sigma})^{-1/2}(\boldsymbol{I} + \tilde{m}^p(-\lambda;\rho)\boldsymbol{\Sigma}_\rho)^{-1}(\tilde{m}_\rho(-\lambda;\rho)\boldsymbol{\Sigma}_\rho + \tilde{m}^p(-\lambda;\rho)\boldsymbol{\Sigma}_\rho')(\boldsymbol{I} + \tilde{m}^p(-\lambda;\rho)\boldsymbol{\Sigma}_\rho)^{-1}(\boldsymbol{I} + \rho\boldsymbol{\Sigma})^{-1/2},\end{aligned}$$

where we recall that $\frac{d\boldsymbol{A}^{-1}}{d\rho} = -\boldsymbol{A}^{-1}\frac{d\boldsymbol{A}}{d\rho}\boldsymbol{A}^{-1}$.

We now calculate $\tilde{m}_\rho(z;\rho)$ and $\boldsymbol{\Sigma}'_\rho$:

$$\begin{aligned}
&-\frac{\tilde{m}_\rho(z;\rho)}{\tilde{m}^2(z;\rho)}\\
=&\frac{1}{n_2}\operatorname{tr}(\boldsymbol{\Sigma}'_\rho(\boldsymbol{I}+\tilde{m}^p(z;\rho)\boldsymbol{\Sigma}_\rho)^{-1}) - \frac{1}{n_2}\operatorname{tr}(\boldsymbol{\Sigma}_\rho(\boldsymbol{I}+\tilde{m}^p(z;\rho)\boldsymbol{\Sigma}_\rho)^{-1}(\tilde{m}_\rho(z;\rho)\boldsymbol{\Sigma}_\rho+\tilde{m}^p(z;\rho)\boldsymbol{\Sigma}'_\rho)(\boldsymbol{I}+\tilde{m}^p(z)\boldsymbol{\Sigma}_\rho)^{-1})
\end{aligned}$$

Therefore, $\tilde{m}_\rho(z;\rho)$ has the closed form

$$\begin{aligned}
&\tilde{m}_\rho(z;\rho)\left(-\frac{1}{(\tilde{m}^p(z))^2}+\frac{1}{n_2}\operatorname{tr}(\boldsymbol{\Sigma}_\rho(\boldsymbol{I}+\tilde{m}^p(z;\rho)\boldsymbol{\Sigma}_\rho)^{-1}\boldsymbol{\Sigma}_\rho(\boldsymbol{I}+\tilde{m}^p(z;\rho)\boldsymbol{\Sigma}_\rho)^{-1})\right)\\
=&\frac{1}{n_2}\operatorname{tr}(\boldsymbol{\Sigma}'_\rho(\boldsymbol{I}+\tilde{m}^p(z;\rho)\boldsymbol{\Sigma}_\rho)^{-1}) - \frac{1}{n_2}\operatorname{tr}(\boldsymbol{\Sigma}_\rho(\boldsymbol{I}+\tilde{m}^p(z;\rho)\boldsymbol{\Sigma}_\rho)^{-1}\tilde{m}^p(z;\rho)\boldsymbol{\Sigma}'_\rho(\boldsymbol{I}+\tilde{m}^p(z;\rho)\boldsymbol{\Sigma}_\rho)^{-1}), \qquad (27)
\end{aligned}$$

and the derivative of $\boldsymbol{\Sigma}_\rho$ reads

$$\boldsymbol{\Sigma}'_\rho = -\frac{1}{2}(\boldsymbol{I}+\rho\boldsymbol{\Sigma})^{-3/2}\boldsymbol{\Sigma}^2(\boldsymbol{I}+\rho\boldsymbol{\Sigma})^{-1/2} - \frac{1}{2}(\boldsymbol{I}+\rho\boldsymbol{\Sigma})^{-1/2}\boldsymbol{\Sigma}(\boldsymbol{I}+\rho\boldsymbol{\Sigma})^{-3/2}\boldsymbol{\Sigma}.$$

We simplify equations at $\rho = 0$ and consider $z = -\lambda$:

$$\begin{aligned}
&\frac{d}{d\rho}\left\{\lambda(\hat{\boldsymbol{\Sigma}}_2+\lambda(\boldsymbol{I}+\rho\boldsymbol{\Sigma})^{-1})\right\}\Big|_{\rho=0}\\
=&-\frac{1}{2}\boldsymbol{\Sigma}(\boldsymbol{I}+\tilde{m}^p(-\lambda)\boldsymbol{\Sigma})^{-1} - \frac{1}{2}(\boldsymbol{I}+\tilde{m}^p(-\lambda)\boldsymbol{\Sigma})^{-1}\boldsymbol{\Sigma} - (\boldsymbol{I}+\tilde{m}^p(-\lambda)\boldsymbol{\Sigma})^{-1}(\tilde{m}^p(-\lambda)\boldsymbol{\Sigma}-\tilde{m}^p(-\lambda)\boldsymbol{\Sigma}^2)(\boldsymbol{I}+\tilde{m}^p(-\lambda)\boldsymbol{\Sigma})^{-1}
\end{aligned}$$

with

$$\tilde{m}^p(-\lambda) = \left(\lambda+\frac{1}{n_2}\operatorname{tr}(\boldsymbol{\Sigma}(\boldsymbol{I}+\tilde{m}^p(-\lambda)\boldsymbol{\Sigma})^{-1})\right)^{-1}$$

and

$$\begin{aligned}
&\tilde{m}_\rho(-\lambda)\left(-\frac{1}{(\tilde{m}^p(-\lambda))^2}+\frac{1}{n_2}\operatorname{tr}(\boldsymbol{\Sigma}(\boldsymbol{I}+\tilde{m}^p(-\lambda)\boldsymbol{\Sigma})^{-1}\boldsymbol{\Sigma}(\boldsymbol{I}+\tilde{m}^p(-\lambda)\boldsymbol{\Sigma})^{-1})\right)\\
=&-\frac{1}{n_2}\operatorname{tr}(\boldsymbol{\Sigma}^2(\boldsymbol{I}+\tilde{m}^p(-\lambda)\boldsymbol{\Sigma})^{-1}) + \frac{1}{n_2}\operatorname{tr}(\boldsymbol{\Sigma}(\boldsymbol{I}+\tilde{m}^p(-\lambda)\boldsymbol{\Sigma})^{-1}\tilde{m}^p(-\lambda)\boldsymbol{\Sigma}^2(\boldsymbol{I}+\tilde{m}^p(-\lambda)\boldsymbol{\Sigma})^{-1}),
\end{aligned}$$

Substitute back to (26) and we have the asymptotic limit of the bias:

$$\begin{aligned}
&B_{\boldsymbol{X}}(\hat{\gamma}_1^{\texttt{calib}}) \xrightarrow{\text{a.s.}}\\
&+\frac{1-\kappa}{2}\operatorname{tr}(\boldsymbol{\Pi}_{\boldsymbol{V}}\boldsymbol{\Sigma}(\boldsymbol{I}+\tilde{m}^p(-\lambda)\boldsymbol{\Sigma})^{-1})\\
&+\frac{1-\kappa}{2}\operatorname{tr}(\boldsymbol{\Pi}_{\boldsymbol{V}}(\boldsymbol{I}+\tilde{m}^p(-\lambda)\boldsymbol{\Sigma})^{-1}\boldsymbol{\Sigma})\\
&+(1-\kappa)\operatorname{tr}(\boldsymbol{\Pi}_{\boldsymbol{V}}(\boldsymbol{I}+\tilde{m}^p(-\lambda)\boldsymbol{\Sigma})^{-1}(\tilde{m}_\rho(-\lambda)\boldsymbol{\Sigma}-\tilde{m}^p(-\lambda)\boldsymbol{\Sigma}^2)(\boldsymbol{I}+\tilde{m}^p(-\lambda)\boldsymbol{\Sigma})^{-1}).
\end{aligned}$$

Further notice that projection onto the column space of $\boldsymbol{V}$ rules out the effects of $\boldsymbol{\Sigma}_s$ on the bias, so we could further simplify the bias term to:

$$\begin{aligned}
&B_{\boldsymbol{X}}(\hat{\gamma}_1^{\texttt{calib}}) \xrightarrow{\text{a.s.}}\\
&+\frac{1-\kappa}{2}\operatorname{tr}(\boldsymbol{\Sigma}_c(\boldsymbol{I}+\tilde{m}(-\lambda)\boldsymbol{\Sigma}_c)^{-1})\\
&+\frac{1-\kappa}{2}\operatorname{tr}((\boldsymbol{I}+\tilde{m}(-\lambda)\boldsymbol{\Sigma}_c)^{-1}\boldsymbol{\Sigma}_c)\\
&+(1-\kappa)\operatorname{tr}((\boldsymbol{I}+\tilde{m}(-\lambda)\boldsymbol{\Sigma}_c)^{-1}(\tilde{m}_\rho(-\lambda)\boldsymbol{\Sigma}_c-\tilde{m}(-\lambda)\boldsymbol{\Sigma}_c^2)(\boldsymbol{I}+\tilde{m}(-\lambda)\boldsymbol{\Sigma}_c)^{-1}).
\end{aligned}$$

This concludes the first part of the proof. Note that in the main theorem we write trace as integral forms.

There are two variance term related to $\hat{\gamma}_1^{calib}$. The first term is the variance contributed by ϵ_1:

$$V_{\epsilon_1}(\hat{\gamma}_1^{calib}) = \frac{\lambda^2\sigma^2}{n_2}\operatorname{tr}((\hat{\boldsymbol{\Sigma}}_2+\lambda\boldsymbol{I})^{-1}\boldsymbol{\Sigma}(\hat{\boldsymbol{\Sigma}}_2+\lambda\boldsymbol{I})^{-1}\hat{\boldsymbol{U}}(\hat{\boldsymbol{U}}^\top\hat{\boldsymbol{\Sigma}}_1\hat{\boldsymbol{U}})^{-1}\hat{\boldsymbol{U}}^\top)\,,$$

which we can bound using Lemma 2. We also have variance contributed by ϵ_2:

$$\begin{aligned} V_{\epsilon_2}(\hat{\gamma}_1^{calib}) =& \frac{\sigma^2}{n_2}\operatorname{tr}(\hat{\boldsymbol{\Sigma}}_2(\hat{\boldsymbol{\Sigma}}_2+\lambda\boldsymbol{I})^{-2}\boldsymbol{\Sigma}) \\ =& \frac{\sigma^2 p}{n_2}\left(\frac{1}{p}\operatorname{tr}(\boldsymbol{\Sigma}(\hat{\boldsymbol{\Sigma}}_2+\lambda\boldsymbol{I})^{-1}) - \frac{\lambda}{p}\operatorname{tr}(\boldsymbol{\Sigma}(\hat{\boldsymbol{\Sigma}}_2+\lambda\boldsymbol{I})^{-2})\right) \end{aligned} \tag{28}$$

We will again use generalized MP law to find the asymptotic deterministic limit of this variance term. By deterministic equivalence, we know that $\frac{1}{p}\operatorname{tr}(\boldsymbol{\Sigma}(\hat{\boldsymbol{\Sigma}}_2+\lambda\boldsymbol{I})^{-1})$ and $\frac{1}{p}\operatorname{tr}(\boldsymbol{\Sigma}\bar{\boldsymbol{Q}}(-\lambda)) = \frac{1}{p}\operatorname{tr}(\boldsymbol{\Sigma}(\lambda\tilde{m}^p(-\lambda)\boldsymbol{\Sigma}+\lambda\boldsymbol{I})^{-1})$ have the same asymptotic limit. To figure out $\tilde{m}^p(-\lambda)$ in (13), we rewrite (14) and (15) as

$$\frac{1}{z\tilde{m}^p(z)} + 1 = 2c\cdot\frac{1}{p}\operatorname{tr}(\boldsymbol{\Sigma}(z\boldsymbol{I}+z\tilde{m}^p(z)\boldsymbol{\Sigma})^{-1}) \tag{29}$$

and

$$\frac{1}{z\tilde{m}(z)} + 1 = 2c\int\frac{t\nu(dt)}{z+z\tilde{m}(z)t}\,. \tag{30}$$

By observing (29) and (30), we arrive at

$$\frac{1}{p}\operatorname{tr}(\boldsymbol{\Sigma}(\hat{\boldsymbol{\Sigma}}_2+\lambda\boldsymbol{I})^{-1}) \xrightarrow{a.s.} \frac{1}{2c}\left(\frac{1}{\lambda\tilde{m}(-\lambda)}-1\right). \tag{31}$$

We will apply the derivative trick to deal with $\frac{1}{p}\operatorname{tr}(\boldsymbol{\Sigma}(\hat{\boldsymbol{\Sigma}}_2+\lambda\boldsymbol{I})^{-2})$. Note that,

$$\frac{1}{p}\operatorname{tr}(\boldsymbol{\Sigma}(\hat{\boldsymbol{\Sigma}}_2+\lambda\boldsymbol{I})^{-2}) = -\frac{1}{p}\frac{d}{d\lambda}\left\{\operatorname{tr}(\boldsymbol{\Sigma}(\hat{\boldsymbol{\Sigma}}_2+\lambda\boldsymbol{I})^{-1})\right\},$$

as a result,

$$\frac{1}{p}\operatorname{tr}(\boldsymbol{\Sigma}(\hat{\boldsymbol{\Sigma}}_2+\lambda\boldsymbol{I})^{-2}) \xrightarrow{a.s.} -\frac{d}{d\lambda}\left\{\frac{1}{2c}\left(\frac{1}{\lambda\tilde{m}(-\lambda)}-1\right)\right\}. \tag{32}$$

Substituting (31) and (32) into (28), with

$$\tilde{m}_z(-\lambda)\left(\frac{1}{\tilde{m}^2(-\lambda)} - \frac{1}{n_2}\operatorname{tr}(\boldsymbol{\Sigma}(\boldsymbol{I}+\tilde{m}(-\lambda)\boldsymbol{\Sigma})^{-1}\boldsymbol{\Sigma}(\boldsymbol{I}+\tilde{m}(-\lambda)\boldsymbol{\Sigma})^{-1})\right) = 1\,,$$

we have (12) and conclude the second part of our proof.

The remaining task is to identify the limit of the cross-bias term $B_{\boldsymbol{X}}(\hat{\gamma}_1^{\texttt{calib}},\hat{\gamma}_2^{\texttt{calib}})$. By applying the same steps as in (22)-(24) and substituting into (16) for $\hat{\gamma}^{\texttt{CPCR}} = \frac{\hat{\gamma}_1^{\texttt{calib}}+\hat{\gamma}_2^{\texttt{calib}}}{2}$, we will have some risks terms for $\hat{\gamma}_1^{\texttt{calib}}$ and $\hat{\gamma}_2^{\texttt{calib}}$, respectively, and also the cross-bias term $B_{\boldsymbol{X}}(\hat{\gamma}_1^{\texttt{calib}},\hat{\gamma}_2^{\texttt{calib}})$ (It is easy to see the cross-variance terms vanish due to independence). More specifically,

$$B_{\boldsymbol{X}}(\hat{\gamma}_1^{\texttt{calib}},\hat{\gamma}_2^{\texttt{calib}}) = \frac{1}{2}\lambda^2(1-\kappa)\operatorname{tr}(\boldsymbol{\Pi}_{\boldsymbol{V}}(\hat{\boldsymbol{\Sigma}}_2+\lambda\boldsymbol{I})^{-1}\boldsymbol{\Sigma}(\hat{\boldsymbol{\Sigma}}_1+\lambda\boldsymbol{I})^{-1})\,.$$

According to how we split the data, $\hat{\boldsymbol{\Sigma}}_1$ and $\hat{\boldsymbol{\Sigma}}_2$ are independent. We treat $(\hat{\boldsymbol{\Sigma}}_1+\lambda\boldsymbol{I})^{-1}$ as a deterministic matrix when applying deterministic equivalent for $(\hat{\boldsymbol{\Sigma}}_2+\lambda\boldsymbol{I})^{-1}$, and repeat for $(\hat{\boldsymbol{\Sigma}}_1+\lambda\boldsymbol{I})^{-1}$. Finally,

$$B_{\boldsymbol{X}}(\hat{\gamma}_1^{\texttt{calib}},\hat{\gamma}_2^{\texttt{calib}}) \xrightarrow{a.s.} \frac{1}{2}(1-\kappa)\operatorname{tr}((\boldsymbol{I}+\tilde{m}(-\lambda)\boldsymbol{\Sigma}_c)^{-2}\boldsymbol{\Sigma}_c)\,.$$

Again in the main theorem we represent trace using integral, and this concludes our proof for Theorem 4.

C.2 Proof of Theorem 2: Optimal λ

The minimum risk is achieved when $\frac{d}{d\lambda}\mathcal{R} = 0$ with the corresponding optimal λ^*. It is straightforward to obtain $B'_{\boldsymbol{X}}(\lambda)$ and $V'(\lambda)$ stated in Theorem 5. Here, we just provide the explicit forms of $\tilde{m}_{\rho z}(\cdot)$ and $\tilde{m}_{zz}(\cdot)$. Recall from the previous proof:

$$\tilde{m}_z(-\lambda)\left(\frac{1}{\tilde{m}^2(-\lambda)} - \frac{1}{n_2}\operatorname{tr}(\boldsymbol{\Sigma}(\boldsymbol{I}+\tilde{m}(-\lambda)\boldsymbol{\Sigma})^{-1}\boldsymbol{\Sigma}(\boldsymbol{I}+\tilde{m}(-\lambda)\boldsymbol{\Sigma})^{-1})\right) = 1\,,$$

and denote $G(-\lambda) = \frac{1}{\tilde{m}^2(-\lambda)} - \frac{1}{n_2}\operatorname{tr}(\boldsymbol{\Sigma}\boldsymbol{A}^{-1}\boldsymbol{\Sigma}\boldsymbol{A}^{-1})$ and $\boldsymbol{A} = \boldsymbol{I} + \tilde{m}(-\lambda)\boldsymbol{\Sigma}$. Therefore,

$$\tilde{m}_{zz}(-\lambda) = \frac{G'(-\lambda)}{G^3(-\lambda)}\,,$$

with $G'(-\lambda) = \frac{2}{\tilde{m}^3(-\lambda)} - \frac{2}{n_2}\operatorname{tr}(\boldsymbol{\Sigma}\boldsymbol{A}^{-1}\boldsymbol{\Sigma}\boldsymbol{A}^{-1}\boldsymbol{\Sigma}\boldsymbol{A}^{-1})$.

Also recall:

$$\begin{aligned}
&\tilde{m}_\rho(-\lambda)\left(-\frac{1}{\tilde{m}^2(-\lambda)} + \frac{1}{n_2}\operatorname{tr}(\boldsymbol{\Sigma}(\boldsymbol{I}+\tilde{m}(-\lambda)\boldsymbol{\Sigma})^{-1}\boldsymbol{\Sigma}(\boldsymbol{I}+\tilde{m}(-\lambda)\boldsymbol{\Sigma})^{-1})\right)\\
=&-\frac{1}{n_2}\operatorname{tr}(\boldsymbol{\Sigma}^2(\boldsymbol{I}+\tilde{m}(-\lambda)\boldsymbol{\Sigma})^{-1}) + \frac{1}{n_2}\operatorname{tr}(\boldsymbol{\Sigma}(\boldsymbol{I}+\tilde{m}(-\lambda)\boldsymbol{\Sigma})^{-1}\tilde{m}(-\lambda)\boldsymbol{\Sigma}^2(\boldsymbol{I}+\tilde{m}(-\lambda)\boldsymbol{\Sigma})^{-1})\,,
\end{aligned}$$

and denote $H(-\lambda) = -\frac{1}{n_2}\operatorname{tr}(\boldsymbol{\Sigma}^2\boldsymbol{A}^{-1}) + \frac{\tilde{m}(-\lambda)}{n}\operatorname{tr}(\boldsymbol{\Sigma}\boldsymbol{A}^{-1}\boldsymbol{\Sigma}^2\boldsymbol{A}^{-1})$. Therefore,

$$\tilde{m}_{\rho z}(-\lambda) = \frac{GH' - HG'}{G^3(-\lambda)}\,,$$

with $H'(-\lambda) = -\frac{2}{n_2}\operatorname{tr}(\boldsymbol{\Sigma}^2\boldsymbol{A}^{-1}\boldsymbol{\Sigma}\boldsymbol{A}^{-1}) + \frac{2\tilde{m}(-\lambda)}{n_2}\operatorname{tr}(\boldsymbol{\Sigma}^2\boldsymbol{A}^{-1}\boldsymbol{\Sigma}\boldsymbol{A}^{-1}\boldsymbol{\Sigma}\boldsymbol{A}^{-1})$.

D ADDITIONAL EXPERIMENT DETAILS [1]

D.1 Kernel Regression with Nystrom Features

In this section, we elaborate on the settings for the numerical experiments in Section 5.2 that are omitted in the main paper. We choose 5 datasets from the UCI Machine Learning repository (Asuncion et al., 2007):

1. D1: Parkinson's disease detection dataset (Little et al., 2008) that predicts Parkinson's disease. The input variables are the voice measurement from subjects, and the output is the status indicator about the disease.
2. D2: Energy efficiency dataset (Tsanas and Xifara, 2012) that predicts the energy performance of residential building. The input variables are the features of building like orientation and relative compactness, and the response value is the energy load of the building.
3. D3: Istanbul stock exchange datasets (Akbilgic et al., 2014) that predicts the index returns. Similar to the origin paper, we adopt the ISE100 index as the response variable and the rest of the stocks market as input variables.
4. D4: Concrete slump test (Yeh, 2007) that predicts slump flow phenomenon. In this regression task, we choose compressive strength as response variable, and the features like cement and water as input variables.
5. D5: QSAR aquatic toxicity dataset (Cassotti et al., 2014) that predicts the aquatic toxicity of different organic molecules. The input variables are the molecular descriptors, and the output is the toxicity.

We use the Nystrom method (Yang et al., 2012) to construct nonlinear features based on raw input features. More specifically, for raw features on two input samples x_i and x_j, we calculate the radial basis function (RBF) as $\kappa(x_i, x_j) = \exp(-||x_i - x_j||^2/(2\sigma^2))$, where σ is a length-scale parameter. The kernel Gram matrix $\widehat{K} = [\kappa(\widehat{x}_i, \widehat{x}_j)]_{m\times m}$ is then constructed by calculating the RBF function on a random subset of m samples. Then, we adopt the Nystrom approximation as $\widehat{K}_r = K_b \widehat{K}^+ K_b^T$, in which $K_b = [\kappa(x_i, \widehat{x}_j)]_{N\times m}$, and $\widehat{K}^+$ is the pseudo inverse of the kernel Gram matrix $\widehat{K}$. Cholesky decomposition on $\widehat{K}_r$ would yield the Nystrom features used in regression analysis. The number of instances, raw feature dimensions, and Nystrom feature dimensions of datasets are summarized in Table 3.

	# Raw Features	# Instances	# Nystrom Features
D1	22	197	200
D2	8	768	800
D3	9	536	600
D4	10	103	200
D5	8	546	600

Table 3: Summary of dataset characteristics

D.2 Classification with Vision Foundation Model (Additional Details)

Data preprocessing and feature extraction. We conduct experiments on the PACS benchmark, which comprises four domains (Photo, Art Painting, Cartoon, Sketch) and seven semantic classes. For each domain, we use pre-extracted image representations obtained from DINOv3 ConvNeXt encoders. Depending on the backbone size, the feature dimension ranges from 768 to 1536; for example, ConvNeXt-tiny yields 768-dimensional embeddings and ConvNeXt-large yields 1536-dimensional embeddings.

Train/test protocol. To emulate an extreme overparameterized regime, we perform a stratified random split within each domain, retaining only 5% of labeled samples for training and using the remaining 95% for evaluation. Across all domains, this results in approximately 500 training samples in total and nearly 10^4 evaluation samples. All domains are pooled together during both training and evaluation; no domain is held out.

[1] All code used in this work is available at `https://github.com/florencewuyixuan/CPCR`.

Label noise model. We inject synthetic label noise into the training set by corrupting a fixed fraction (20%) of labels independently. For each selected sample, the ground-truth label is replaced by a uniformly sampled incorrect class label (i.e., drawn uniformly from the remaining classes). This corresponds to a uniform label noise model (Zhu et al., 2024).

Model training. All methods are implemented as linear classifiers on top of pre-extracted, frozen foundation model features. For logistic regression (LR), we use an ℓ_2-regularized multinomial model with the `lbfgs` solver, no intercept term, and a maximum of 200 iterations. The regularization parameter C is selected from the grid $\{10^{-10}, 10^{-2}, 10^{-1}, 1, 10, 100, 10^{10}\}$.

For PCR and CPCR, we first compute principal components via singular value decomposition. Importantly, the PCA subspace is estimated using all available feature vectors (including both training and evaluation samples), reflecting a semi-supervised setting where unlabeled test data are availble during training. We then project the training data onto the top-r components and fit the same logistic regression model. Unless otherwise specified, we fix $r = 8$.

CPCR further applies a cross-fitted debiasing procedure with ridge regularization. The ridge parameter is coupled with the LR regularization through $\lambda = 1/C$, and we report the best performance over this shared grid.

Implementation details. All experiments are conducted with fixed random seeds to ensure reproducibility. We enforce deterministic behavior in both NumPy and PyTorch (including CUDA backends when available). Training is lightweight and completes within seconds on a single GPU (e.g., NVIDIA RTX 3060).

Model selection and reporting. For each method, we report the best test accuracy over the hyperparameter grid. Results are averaged over five independent random splits, and we report mean and standard deviation.