Leveraging semantic similarity for experimentation with AI-generated treatments

Lei Shi
University of California, Berkeley
Berkeley, CA 94720
leishi@berkeley.edu

David Arbour
Adobe Research
San Jose, CA 95110
arbour@adobe.com

Raghavendra Addanki
Adobe Research
San Jose, CA 95110
raddanki@adobe.com

Ritwik Sinha
Adobe Research
San Jose, CA 95110
risinha@adobe.com

Avi Feller
University of California, Berkeley
Berkeley, CA 94720
afeller@berkeley.edu

Abstract

Large Language Models (LLMs) enable a new form of digital experimentation where treatments combine human and model-generated content in increasingly sophisticated ways. The main methodological challenge in this setting is representing these high-dimensional treatments without losing their semantic meaning or rendering analysis intractable. Here we address this problem by focusing on learning low-dimensional representations that capture the underlying structure of such treatments. These representations enable downstream applications such as guiding generative models to produce meaningful treatment variants and facilitating adaptive assignment in online experiments. We propose double kernel representation learning, which models the causal effect through the inner product of kernel-based representations of treatments and user covariates. We develop an alternating-minimization algorithm that learns these representations efficiently from data and provide convergence guarantees under a low-rank factor model. As an application of this framework, we introduce an adaptive design strategy for online experimentation and demonstrate the method's effectiveness through numerical experiments.

1 Introduction

Randomized experiments are a cornerstone of measuring the efficacy of content (e.g., messaging, imagery) in modern industry (e.g., Sun et al., 2021; Grimmer and Woolley, 2014; Leung et al., 2017) and social science (e.g., Gerber et al., 2018; Green et al., 2023; Salovey and Williams-Piehota, 2004) contexts. Historically, generating the content itself has been the core challenge. The advent of Large Language Models (LLMs) and other generative models involving text, images, and videos (Gołąb-Andrzejak, 2023; Baek, 2023; Kumar and Kapoor, 2024) in recent years has dramatically decreased this design burden. By integrating human creativity with model-generated content, generative models are transforming digital experimentation and streamlining treatment generation.

The ability to generate many content variants for experiments, however, has come with new statistical challenges, especially since the corresponding experiment populations have largely stayed fixed. As a result, practitioners often face the choice of either running underpowered experiments or leaving many treatment variations unexamined. In this paper, we explore an alternative: to improve statistical efficiency by explicitly incorporating semantic similarity from treatment embeddings (numeric representations of the treatments). This raises two key technical challenges: (i) Effectively leveraging

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

semantic information for better estimation and experimental designs, and (ii) Developing a framework that appropriately captures the relationship between generated variants and user attributes. Dealing with these challenges can facilitate many downstream applications, such as guiding generative models to generate treatments and assisting adaptive treatment assignment in online experiments.

In this work, we propose a kernel-based representation learning approach that addresses these challenges. Our framework integrates both treatment embeddings and user covariate information to estimate the causal effects of LLM-generated variants. Specifically, our core contributions are:

1. A double-kernel representation learning framework in which the treatment effect is the inner product of low-dimensional representations of treatment embeddings and user covariates.
2. A computationally efficient alternating minimization-style algorithm to learn these unknown representations. This naturally extends to an adaptive algorithm that assigns treatments to users in an online manner.
3. Theoretical guarantees, including convergence rates for treatment effect estimation and a sublinear regret bounds of an algorithm for the adaptive experimentation setting.

2 Related Work and Background

Our work builds on several key threads.

Causal inference with structured treatments. There is an extensive literature on causal inference with text; see Feder et al. (2022) for a comprehensive survey. Prior work has largely focused on the challenging problem of learning latent treatments from images or text (Fong and Grimmer, 2023; Egami et al., 2022). We sidestep several of these issues by taking the generated texts as fixed treatments and only focusing on using semantic information to improve efficiency. Many recent works have started exploring textual causal inference with LLMs, such as Imai and Nakamura (2024); Tierney et al. (2025). However, they are not touching the problem of personalized treatment effect estimation and adaptive experimentation.

High-dimensional and continuous treatments. Our work builds most directly on recent results on heterogeneous treatment effects with high-dimensional and continuous treatments. Kaddour et al. (2021) extended the R-learner approach of Nie and Wager (2021) to a setting with unstructured treatments. Liu et al. (2024) studied continuous treatment effect estimation by modeling dependences among treatment and responses using covariate embeddings from generative models. Singh et al. (2024) proposed estimators based on kernel ridge regression for nonparametric causal functions such as dose, heterogeneous, and incremental response curves.

More generally, there has been substantial recent work on experimentation with high-dimensional or continuous treatments, especially in the structured bandit problems, such as linear bandits (Rusmevichientong and Tsitsiklis, 2010; Hao et al., 2020; Oh et al., 2021; Komiyama and Imaizumi, 2024), generalized linear bandits (Filippi et al., 2010; Li et al., 2017), low-rank bandits (Lu et al., 2021; Jun et al., 2019), among others.

Matrix completion for causal inference. There is a substantial literature on decomposing a function (e.g., treatment effect) based on matrix factorization, such as Abernethy et al. (2006); Fithian and Mazumder (2013); Mao et al. (2019); Zhou et al. (2012); Athey et al. (2021). Most relevant for our work, Jin et al. (2022) propose low-rank matrix completion when covariate information is available. Importantly, Jin et al. (2022) only have covariate information available for the treatment-user pair, which differs from our setting in which we have both treatment embeddings and user attributes available.

Kernel methods. We build on the expansive causal inference literature that leveraging kernel methods: kernel-based methods that incorporate treatment embeddings for experimental design and analysis offer several benefits. First, kernel-based methods allow for flexible response surface modeling, $f \in \mathbb{H}_{\mathcal{K}}$, going beyond simpler parametric formulations such as linear models (Rusmevichientong and Tsitsiklis, 2010; Lu et al., 2010). Second, kernel-based methods provide adaptive design frameworks through tools including kernel bandits (Srinivas et al., 2009; Valko et al., 2013) and Bayes optimization (Frazier, 2018; Martinez-Cantin, 2014; Ignatiadis and Wager, 2019; Dimmery et al., 2019). Third, kernel-based methods have well-established statistical guarantees, including prediction/estimation error analysis (Mendelson and Neeman, 2010; Wainwright, 2019) and regret analysis (Srinivas et al., 2009; Frazier, 2018; Chowdhury and Gopalan, 2017; Valko et al., 2013; Vakili et al., 2023).

3 Problem description

3.1 Setup

Notation. We first introduce notation. Throughout, for an integer k, $[k]$ denotes the set $\{1, \dots, k\}$. For a matrix $\mathbf{\Gamma}$, $\|\mathbf{\Gamma}\|_F$ is the Frobenius norm, $\|\mathbf{\Gamma}\|_*$ is the nuclear norm, and $\|\mathbf{\Gamma}\|_\infty$ is the element-wise infinity norm. $\rho_{\min}(\mathbf{\Gamma})$ and $\rho_{\max}(\mathbf{\Gamma})$ denote the smallest and largest singular values, respectively.

Causal inference problem. Evaluating many content variants generated by LLMs creates challenges for classical A/B testing. We formalize these challenges under a causal inference framework. For a user with covariate $x \in \mathcal{X}$, we posit the following structural model for the potential outcome under the treatment $z \in \mathcal{Z}$:

$$y(z) = f(z, x) + \epsilon, \quad z \in \mathcal{Z}, \quad x \in \mathcal{X}. \tag{1}$$

Here f is an unknown smooth function, and ϵ is mean-zero random noise. $\mathcal{Z}$ and $\mathcal{X}$ can be either finite or continuous spaces, representing the treatment and user covariate space, respectively. The structural model (1) implicitly encodes SUTVA: (i) no interference between units, meaning one units' treatments or outcomes do not affect another unit's potential outcomes; and (ii) no hidden version of treatments, meaning the realized outcome is the potential outcome at the assigned treatment.

For the estimation problem, we are interested in comparing a new treatment z with a control variant z_0, for a user with covariates x. The conditional average treatment effect is then defined as:

$$\tau(z, x) = f(z, x) - f(z_0, x).$$

When z is binary, i.e., $z \in \{z_0, z_1\}$, $\tau(z, x)$ is the standard conditional average treatment effect.

Learning $\tau(z, x)$ efficiently from data is crucial for both estimation and experimentation. For traditional A/B testing, the treatment space $\mathcal{Z}$ involves a finite number of variants, which can be fully explored by assigning each element in $\mathcal{Z}$ to a sufficient number of users. However, treating variants as separate discrete elements in LLM-powered experiments leads to two issues: (i) experiments are hugely underpowered since it is difficult to fully explore the treatment space $\mathcal{Z}$ due to the large number of variants; (ii) the experiments ignore the semantic information in the variants, which encodes similarity between arms and could deliver important information about the outcome function. The two questions motivate the idea of leveraging treatment embeddings to explore the similarity between treatment arms and pool information across variants to improve statistical efficiency. We will now describe each of these problems in turn.

3.2 Warmup: learning with treatment embeddings

To illustrate the use of embeddings, we begin with a simple setup in which the outcome function is homogeneous among users, i.e., $f(z, x) = f(z)$ is only a function of the treatments, not the user covariates; we relax this below. In a typical A/B testing framework, z is a discrete set, which makes estimation difficult when there are a large number of treatments. Existing approaches to improve efficiency using shrinkage (Dimmery et al., 2019; Ignatiadis and Wager, 2019) help alleviate this but still face fundamental limits because of the discrete treatment space. However, in our setting, each generated hypothesis (or treatment) is typically associated with an embedding that encodes important information. In the generative setting, we can easily obtain such an embedding by, e.g., using the last layer of a deep neural network that generated the treatment. By representing the treatment space $\boldsymbol{Z}$ as an embedding space, with each treatment lying in $\mathbb{R}^p$ (possibly with large p), instead of a space of unrelated discrete elements, we avoid some of the difficulties associated with high cardinality treatments by implicitly pooling observations across treatments that are close in embedding space.

While conceptually straightforward, the continuous parameterization raises several estimation challenges. We tackle those using kernel methods, which is easily adapted to measuring similarity between treatments over the embedding space $\mathcal{Z}$. In particular, we apply kernel ridge regression (based on a kernel $\mathcal{K}$) to obtain an estimator $\widehat{f}$ for the outcome response curve. Given sample (z_i, y_i), we fit the following penalized regression:

$$\widehat{f} = \arg\min_{f \in \mathbb{H}_{\mathcal{K}}} \frac{1}{2n} \sum_{i=1}^{n} (y_i - f(z_i))^2 + \lambda_n \|f\|_{\mathbb{H}_{\mathcal{K}}}.$$

We then predict the estimated outcome response given a specific treatment $z \in \mathcal{Z}$ as $\widehat{f}(z)$. See Appendix A for additional details.

3.3 Warmup: Incorporating covariates

Building on this simple case, we can then extend the idea of incorporating treatment embeddings into a general setup that also includes user covariates. In an idealized setting, we would estimate the kernels with access to all $n \times n$ (expected) potential outcomes $\{f(z_k, x_i)\}_{k,i \in [n]}$, For example, a natural approach for this structure is the multi-task Gaussian process framework introduced by Bonilla et al. (2007), which defines $\mathcal{K}_g$ and $\mathcal{K}_h$ as kernel functions that depict the similarity between treatments and individual covariates. In practice, however, the fundamental problem of causal inference makes directly estimating $f(z, x)$ challenging because most values of this function are missing. In the next section, we propose a kernel-based approach that makes progress despite these hurdles.

4 Learning with treatment embeddings and individual covariates

We now present our main approach, which we call *Double Kernel Representation Learning* for treatment effect estimation. In section 4.1, we introduce a factorization that captures the estimation problem as an inner product of representations involving treatment embeddings and user attributes. We describe an alternating minimization style algorithm that recovers these representations. In Section 4.2, we provide convergence guarantees of our estimator constructed from these representations under a fixed basis.

4.1 Double-kernel representation learning

4.1.1 Setup and problem formulation

For binary treatment, Nie and Wager (2021) considered the following reformulated outcome model:

$$y = f(z_0, x) + \mathbf{1}\{z = z_1\} \cdot \tau(x). \tag{2}$$

Define the outcome model $m(x) = \mathbb{E}\{y \mid x\}$ and propensity score $e(x) = \mathbb{E}\{\mathbf{1}\{z = z_1\} \mid x\}$. Nie and Wager (2021) introduced the Robinson Decomposition (Robinson, 1988) and further transformed (2) into the following partial linear model:

$$y = m(x) + (\mathbf{1}\{z = z_1\} - e(x)) \cdot \tau(x).$$

Kaddour et al. (2021) extended this approach to the setting where z is a high-dimensional or continuous treatment by directly assuming $f(z, x)$ in (1) can be factorized as follows:

$$f(z, x) = \boldsymbol{g}(z)^\top \boldsymbol{h}(x),$$

where $\boldsymbol{g} : \mathbb{R}^p \to \mathbb{R}^r$ and $\boldsymbol{h} : \mathbb{R}^q \to \mathbb{R}^r$ are mappings in the real vector space.

Here, we take a slightly different perspective: instead of factorizing $f(z, x)$, we factorize the treatment effect $\tau(z, x)$, similar to Nie and Wager (2021). We posit the following partial linear model for f:

$$f(z, x) - f(z_0, x) = \boldsymbol{g}(z)^\top \boldsymbol{h}(x),$$

which yields:

$$y = f(z_0, x) + \boldsymbol{g}(z)^\top \boldsymbol{h}(x) + \epsilon. \tag{3}$$

The above factorization states that the CATE function has a low-dimensional representation despite the high-dimensional covariate and treatment inputs. Such a structure is common in the literature (see Section 2) and has become foundational for estimation and optimization. Following Wainwright (2019); Rohde and Tsybakov (2011), we can further relax this to be approximately low-dimensional, allowing a sparse set of large signals along with many small signals.

We can further condition on the user-level covariates by reparametrizing the model (3) and taking the conditional expectation with respect to x. Formally, assume the positivity assumption:

$$\mathbb{P}\{z \mid x\} > 0, \text{ for all } z \in \mathcal{Z}, x \in \mathcal{X}.$$

Let $\overline{\boldsymbol{g}}(x) = \mathbb{E}\{\boldsymbol{g}(z) \mid x\}^\top$. Then,

$$m(x) = f(z_0, x) + \overline{\boldsymbol{g}}(x)^\top \boldsymbol{h}(x). \tag{4}$$

Taking the difference between (3) and (4) yields:

$$y = m(x) + (\boldsymbol{g}(z) - \overline{\boldsymbol{g}}(x))^\top \boldsymbol{h}(x) + \epsilon. \tag{5}$$

In the special case where randomization to treatment z is independent of x, $\overline{\boldsymbol{g}}(x) = \overline{\boldsymbol{g}}$ is a constant function; we start with this case and still denote $\boldsymbol{g}(z) - \overline{\boldsymbol{g}}$ as $\boldsymbol{g}(z)$. In addition, we can directly use the residuals $y - m(x)$ for estimation by plugging in an estimated outcome model $m(x)$. For simplicity, we therefore take y to be the residualized outcome here without loss of generality.

4.1.2 Estimation and main result

Building on the above, a natural idea is to solve for $\boldsymbol{g}$ and $\boldsymbol{h}$ via the following double kernel regression:

$$(\widehat{\boldsymbol{g}}, \widehat{\boldsymbol{h}}) = \arg \min_{\{g_l, h_l\}_{l=1}^r} \frac{1}{2n} \sum_{i=1}^n \left\{ y_i - \sum_{l=1}^r g_l(z_i) h_l(x_i) \right\}^2 + \lambda_n \sum_{l=1}^r (\|g_l\|_{\mathcal{K}_g}^2 + \|h_l\|_{\mathcal{K}_h}^2). \tag{6}$$

We then prove the following representer theorem (see Appendix A and D.1):

Theorem 4.1 (Representer theorem)**.** *The optimal solutions $\boldsymbol{g}^\star$ and $\boldsymbol{h}^\star$ to the optimization problem 6 lie in $\mathbb{H}_{\mathcal{K}_g}^S$ and $\mathbb{H}_{\mathcal{K}_h}^S$ respectively.*

The representer theorem immediately suggests that the solutions to $\widehat{\boldsymbol{g}}$ and $\widehat{\boldsymbol{h}}$ have the following form:

$$\widehat{\boldsymbol{g}}(z) = \sum_{i \in [n]} \boldsymbol{U}_i^{\mathrm{R}} \mathcal{K}_g(z, z_i), \quad \widehat{\boldsymbol{h}}(x) = \sum_{i \in [n]} \boldsymbol{V}_i^{\mathrm{R}} \mathcal{K}_h(x, x_i), \tag{7}$$

where the vectors $\boldsymbol{U}_i^{\mathrm{R}}, \boldsymbol{V}_i^{\mathrm{R}} \in \mathbb{R}^r$. For convenience, we capture them using the following matrices: $\boldsymbol{U} = [\boldsymbol{U}_1^{\mathrm{R}}, \ldots, \boldsymbol{U}_n^{\mathrm{R}}]^\top \in \mathbb{R}^{n \times r}$ and $\boldsymbol{V} = [\boldsymbol{V}_1^{\mathrm{R}}, \ldots, \boldsymbol{V}_n^{\mathrm{R}}]^\top \in \mathbb{R}^{n \times r}$, each row representing a feature representation for a unit. Alternatively, we can write $\boldsymbol{U}$ and $\boldsymbol{V}$ in terms of their columns: $\boldsymbol{U} = [\boldsymbol{U}_1^{\mathrm{C}}, \ldots, \boldsymbol{U}_r^{\mathrm{C}}]$ and $\boldsymbol{V} = [\boldsymbol{V}_1^{\mathrm{C}}, \ldots, \boldsymbol{V}_r^{\mathrm{C}}]$. With (7), we have a double RKHS model for f:

$$f(x, t) = \sum_{i \in [n]} \sum_{j \in [n]} \left\langle \boldsymbol{U}_i^{\mathrm{R}}, \boldsymbol{V}_j^{\mathrm{R}} \right\rangle \mathcal{K}_g(z, z_i) \mathcal{K}_h(x, x_i) = \mathcal{K}_g(z, \boldsymbol{z}_{1:n}) \boldsymbol{\Theta} \mathcal{K}_h(x, \boldsymbol{x}_{1:n})^\top,$$

where $\boldsymbol{\Theta} = \boldsymbol{U}\boldsymbol{V}^\top \in \mathbb{R}^{n \times n}$ is a low-rank matrix. We therefore propose the following double RKHS regression:

$$(\widehat{\boldsymbol{U}}, \widehat{\boldsymbol{V}}) = \arg\min_{\boldsymbol{U}, \boldsymbol{V}} \frac{1}{2n} \sum_{k=1}^n \{y_k - \mathcal{K}_g(z_k, \boldsymbol{z}_{1:n}) \boldsymbol{U}\boldsymbol{V}^\top \mathcal{K}_h(x_k, \boldsymbol{x}_{1:n})^\top\}^2 + \lambda_n \sum_{l \in [r]} (\boldsymbol{U}_l^{\mathrm{C}\top} \mathcal{K}_g \boldsymbol{U}_l^{\mathrm{C}} + \boldsymbol{V}_l^{\mathrm{C}\top} \mathcal{K}_h \boldsymbol{V}_l^{\mathrm{C}}). \tag{8}$$

Computationally, (8) can be solved by alternatively minimizing over $\boldsymbol{U}$ and $\boldsymbol{V}$. Concretely, suppose we have $(\widehat{\boldsymbol{U}}^{(t)}, \widehat{\boldsymbol{V}}^{(t)})$ from the t-th iteration. In the $(t+1)$-th round, we update the parameters as follows:

$$\widehat{\boldsymbol{U}}^{(t+1)} = \arg\min_{\boldsymbol{U}} \frac{1}{2n} \sum_{k=1}^n \{y_k - \mathcal{K}_g(z_k, \boldsymbol{z}_{1:n}) \boldsymbol{U}\boldsymbol{V}^{(t)\top} \mathcal{K}_h(x_k, \boldsymbol{x}_{1:n})^\top\}^2 + \lambda_n \sum_{l \in [r]} \boldsymbol{U}_l^{\mathrm{C}\top} \mathcal{K}_g \boldsymbol{U}_l^{\mathrm{C}}, \tag{9}$$

$$\widehat{\boldsymbol{V}}^{(t+1)} = \arg\min_{\boldsymbol{V}} \frac{1}{2n} \sum_{k=1}^n \{y_k - \mathcal{K}_g(z_k, \boldsymbol{z}_{1:n}) \boldsymbol{U}^{(t+1)}\boldsymbol{V}^\top \mathcal{K}_h(x_k, \boldsymbol{x}_{1:n})^\top\}^2 + \lambda_n \sum_{l \in [r]} \boldsymbol{V}_l^{\mathrm{C}\top} \mathcal{K}_h \boldsymbol{V}_l^{\mathrm{C}}. \tag{10}$$

Both (9) and (10) have closed-form solutions when viewed as weighted ridge regression. Moreover, the alternating minimization process guarantees a descending (thus convergent) loss function:

$$L(\widehat{\boldsymbol{U}}^{(t)}, \widehat{\boldsymbol{V}}^{(t)}) \geq L(\widehat{\boldsymbol{U}}^{(t+1)}, \widehat{\boldsymbol{V}}^{(t)}) \geq L(\widehat{\boldsymbol{U}}^{(t+1)}, \widehat{\boldsymbol{V}}^{(t+1)}).$$

Algorithm 1 Double Kernel Learning via Alternating Projection

Input: data (x_i, z_i, y_i), kernel $\mathcal{K}_g, \mathcal{K}_h$, rank r, penalty level λ_n, maximal iteration T, accuracy `tol`

Initialize $\boldsymbol{U}^{(0)}$ and $\boldsymbol{V}^{(0)}$ as random matrices with gaussian elements.

for $t = 0$ **to** T **do**

  Update $\boldsymbol{U}^{(t)} \to \boldsymbol{U}^{(t+1)}$ by solving (9);

  Update $\boldsymbol{V}^{(t)} \to \boldsymbol{V}^{(t+1)}$ by solving (10);

  if $\|\boldsymbol{U}^{(t)} - \boldsymbol{U}^{(t+1)}\|_F / \|\boldsymbol{U}^{(t)}\|_F <$ `tol` and $\|\boldsymbol{V}^{(t)} - \boldsymbol{V}^{(t+1)}\|_F / \|\boldsymbol{V}^{(t)}\|_F <$ `tol` **then**

    Break

  end if

end for

Output: $\boldsymbol{U}^{(t)}, \boldsymbol{V}^{(t)}$

We summarize the procedure in Algorithm 1[1], along with some discussions.

Complexity of Algorithm 1. Algorithm 1 involves a loop that alternates between solving (9) and (10), thus the computational complexity depends on the solver for (9) and (10). In our implementation, we solve (9) and (10) by updating the columns of $\boldsymbol{U}$ and $\boldsymbol{V}$ recursively via ridge regression with a general quadratic form penalty. To run these ridge regressions, we can compute the inverse of the kernel matrices before the loop. With these holdout inverse matrices, we only need $O(N^2)$ to solve each round of the generalized ridge regression. Therefore, in total, the time complexity is

$$O(\underbrace{N^3}_{\text{Compute inversion of kernel matrices}}) + O(\underbrace{T}_{\text{num of loops}} \cdot \underbrace{r}_{\text{Rounds of ridge regressions}} \cdot \underbrace{N^2}_{\text{compute ridge updates}}).$$

Scalability to large-scale datasets. To make the algorithm adaptive to large-scale datasets (especially with large N), we can leverage standard techniques to speed up kernel computation. For example, using the Nyström method (Williams and Seeger, 2000) and its variations can reduce the inversion complexity to linear in N.

Convergence of the algorithm. The optimization performance of alternative minimization is well studied. Since we are more concerned with the statistical properties (see Theorem 4.2) and experimentation performance (see Theorem 5.1) of the proposed method, we refer interested readers to optimization-based discussions in Jain et al. (2013); Chi et al. (2019).

Comparison with a deep-learning based framework. Deep-learning offers an alternative framework for learning CATE from (3) (Kaddour et al., 2021). That approach focuses primarily on prediction, often resulting in difficult-to-interpret representations for the learned treatment and covariates. The corresponding theoretical results are also quite different: as we show in Section 4.2 below, we provide rigorous theoretical guarantees for the estimation accuracy of reconstructing feature representations. By contrast, theoretical results for deep learning-based approaches mainly focus on proving bounds on excess risk as a metric to measure prediction accuracy, but do not provide results on how well the representations themselves are constructed. Finally, deep learning based approaches are computationally expensive relative to the low computational cost of kernel-based methods.

Connection with classical approaches. The proposed double kernel learning method has a deep connection with two classical approaches: (i) kernelized probability matrix factorization, which was introduced by Zhou et al. (2012) and serves as the Gaussian process counterpart of our method; and (ii) low-rank matrix factorization, which explores low-rank signals without a kernel structure and has been discussed extensively in the literature (Recht et al., 2010; Chi et al., 2019). We explore these connections in more depth in Section B of the Appendix.

4.2 Theoretical analysis for fixed bases

For our theoretical analysis, we follow Kaddour et al. (2021) and consider a fixed basis setting. Suppose there are d_1 treatment bases $\mathcal{Z} = \{z^k\}$, which is a set of p-dimensional vectors, and d_2 user bases $\mathcal{X} = \{x^k\}$, which is a set of q dimensional vectors. $\mathcal{Z}$ and $\mathcal{X}$ summarize the information of the treatment to be tested and the pool of candidate users, respectively. Stack the bases into two matrices:

$$\boldsymbol{Z} = [z^1, \dots, z^{d_1}] \in \mathbb{R}^{p \times d_1}, \quad \boldsymbol{X} = [x^1, \dots, x^{d_2}] \in \mathbb{R}^{q \times d_2}.$$

[1]There are other methods in the literature that may be used for solving the program, such as (stochastic) gradient descent (Zhou et al., 2012) and Quasi-Newton methods (Abernethy et al., 2006).

Let (e_z, e_x) be a pair of independent uniform samples in the product set $\mathcal{Z} \times \mathcal{X}$. The observation i is associated with one covariate $x_i = \boldsymbol{X} e_{x,i}$ and one treatment $z_i = \boldsymbol{Z} e_{z,i}$. Based on our discussion in Section 4.1, under the decomposition model (5), there exists a low rank matrix $\boldsymbol{\Theta}^\star$, such that the observed outcome is

$$y_i = m(x_i) + z_i^\top \boldsymbol{\Theta}^\star x_i + \epsilon_i = m(x_i) + e_{z,i}^\top \boldsymbol{Z}^\top \boldsymbol{\Theta}^\star \boldsymbol{X} e_{x,i} + \epsilon_i.$$

Define the new matrix

$$\boldsymbol{\Gamma}^\star = \boldsymbol{Z}^\top \boldsymbol{\Theta}^\star \boldsymbol{X} \in \mathbb{R}^{d_1 \times d_2}, \tag{11}$$

whose rank is at most r if rank$(\boldsymbol{\Theta}^\star) \leq r$. We prove the following theorem:

Theorem 4.2. *Assume that the noise ϵ_i's are i.i.d. and sub-exponential, and the true signal matrix $\boldsymbol{\Gamma}^\star$ has rank at most r and entries bounded by $\alpha^\star$. With probability greater than $1 - c_1' \exp(-c_2' d \log d)$ with $d = d_1 + d_2$, the estimator $\widehat{\boldsymbol{\Gamma}}$ satisfies*

$$\frac{1}{d_1 d_2} \|\widehat{\boldsymbol{\Gamma}} - \boldsymbol{\Gamma}^\star\|_F^2 \leq C \lambda_n^2 r, \quad \textit{where} \quad \lambda_n \asymp \max\left\{\frac{1}{\sqrt{n}} \|\widehat{m}(x) - m(x)\|_2, \sqrt{\frac{d \log d}{n}}\right\}.$$

Theorem 4.2 suggests that $\widehat{\boldsymbol{\Gamma}}$ is consistent in large samples as long as $\lambda_n^2 r \to 0$. When the rank r is fixed, consistency then only requires a vanishing penalization level λ_n. We can then consider each element of λ_n in turn. The left term captures the error in the outcome regression function, $m(x)$. For this term to vanish, we need a consistent estimator $\widehat{m}(x)$ of $m(x)$ in the sense that $(1/\sqrt{n})\|\widehat{m}(x) - m(x)\|_2 \to 0$. This is guaranteed by many statistical/machine learning methods $\widehat{m}$ under the assumption that $m(x)$ belongs to an appropriate function class. For example, if $\widehat{m}(x)$ is fit with LASSO, the loss vanishes at a rate of $\sqrt{s \log p/n}$ where p is the dimension of the features and s is the sparsity level (Bickel et al., 2008). If $\widehat{m}(x)$ is fit via RKHS regression with a Gaussian kernel, the rate is $\sqrt{\log n/n}$ (Wainwright, 2019; Ma et al., 2023). The right term, $\sqrt{d \log d/n}$, captures the cost of low-rank matrix recovery. For this term to vanish, $n \gg dr \log d$, where dr captures the intrinsic dimension of a low-rank matrix (Candes and Plan, 2011). This rate has been verified to be minimax-optimal for matrix completion tasks (Rohde and Tsybakov, 2011; Negahban and Wainwright, 2012; Klopp, 2014). Finally, following Wainwright (2019); Rohde and Tsybakov (2011), we note that it is possible to relax this setup to be approximately low-rank, in the sense that the true signal can admit a set of large signals along with many small signals. We relegate a comprehensive analysis under the approximate low-rank regime to Appendix D.

5 Online Experimentation: An Adaptive Strategy

In this section, we provide an adaptive algorithm for assigning treatments to individuals as they become available in an online manner. We use Theorem 4.2 to justify the performance of an explore-then-commit strategy. Consider a bandit with T rounds, which we divide into two stages. In the first stage, we use T_e rounds to explore the best arm for each user by drawing some samples and learning the outcome response matrix Γ. In the second stage, we keep pulling from the learned best arm for each selected user. Algorithm 2 summarizes this approach.

We prove that the explore-then-commit algorithm will lead to a sublinear regret. To see this, we establish the following theorem:

Theorem 5.1. *Assume the conditions in Theorem 4.2. Also, assume that $m(x)$ lies in the RKHS generated by a Gaussian kernel. The explore-then-commit algorithm has the following sublinear regret bound with probability greater than $1 - c_1' \exp(-c_2' d \log d)$:* Regret $\leq CT^{2/3} d \log d^{1/3} r^{1/3}$, *where $d = d_1 + d_2$.*

We can see that the simple Explore-then-Commit strategy suffices to guarantee a regret with rate $O(T^{2/3})$, which is sublinear. We anticipate that we can use ideas from the literature on low-rank bandits, such as Jun et al. (2019) and Lu et al. (2021), to sharpen this regret bound in terms of the horizon T, though this is outside the scope of the current paper.

6 Experimental Evaluation

In this section, we conduct semi-synthetic experiments based on three open-source datasets. (i) The Upworthy Research Archive (Matias et al., 2021) is an open dataset of thousands of A/B tests of

Algorithm 2 Explore-then-commit

Input: Exploration rounds T_e and exploitation rounds T_c, total rounds T, treatment bases $\mathcal{Z}$, covariates $\mathcal{X}$
```
# Data collection for exploration:
```
for $t = 1$ **to** T_e **do**
  Randomly sample $e_{z,t}$ with replacement and obtain a treatment $z_t = e_{z,t}^\top \boldsymbol{Z}$;
  Randomly sample a $e_{x,t}$ with replacement and obtain a target user with covariate $x_t = e_{x,t}^\top \boldsymbol{X}$;
  Assign the user to treatment z_t and observe outcome response y_t;
end for
```
# Learn outcome responses:
```
Learn conditional mean $\widehat{m}(x)$ with machine learning algorithms;
Learn a outcome response matrix $\widehat{\boldsymbol{\Gamma}}$ by solving (6);
```
# Commit to the best personalized treatment:
```
for $t = T_e + 1$ **to** T **do**
  Randomly sample $e_{x,t}$ with replacement from $\mathcal{X}$;
  Assign the user to treatment z_t which maximizes the estimated outcome response vector $\widehat{\boldsymbol{\Gamma}} e_{x,t}$;
  Collect outcome response y_t;
end for
Output: outcome response matrix $\widehat{\boldsymbol{\Gamma}}$, data (z_t, x_t, y_t).

headlines conducted by Upworthy from January 2013 to April 2015. We use the Exploratory Dataset in The Upworthy Research Archive, which contains headlines and metrics (clicks, impressions, click-through rates, or CTR) from thousands of experiments. (ii) The MIND dataset (Wu et al., 2020) is a benchmark dataset containing traffic from Microsoft for click rate prediction and news recommendation. (iii) The ASOS E-Commerce Dataset[2] is an open-source A/B testing dataset from ASOS in the fashion industry. Due to space constraints, we present a subset of results regarding the Upworthy experiment and MIND dataset, and relegate the remaining results to Section E in the Appendix. The code for the algorithm and replication of the numerical experiments can be found here: `https://github.com/LeiShi-rocks/DKRL-LLM`.

6.1 The Upworthy experiment

Upworthy experimental setup. We choose $|\mathcal{Z}| = 50$ headlines as candidate treatments in a hypothetical experiment. We use the sentence transformer `MiniLM` (Wang et al., 2020) to encode the sampled headlines into sentence embeddings of dimension $p = 384$. Since user-level covariates x are not available in the Upworthy Dataset, we simulate $n = 500$ Gaussian vectors of dimension $q = 200$ as baseline covariates. The outcome of interest is the potential revenue that each user can contribute. When a user with covariate x views the headline z, we model the average (centered) potential revenue generated by this particular user as $f(z, x) = z^\top \boldsymbol{\Theta}^\star x + \epsilon$, where $\boldsymbol{\Theta}^\star \in \mathbb{R}^{p \times q}$ is a matrix with varying ranks and ϵ is additive noise.

Evaluation of the estimation error. We use Monte Carlo simulations to evaluate the train and test error of the proposed method, double kernel representation learning (`DKRL`), and compare with two baseline methods: (i) Structured Intervention Network (`SIN`) by Kaddour et al. (2021), a deep-learning based framework based on a generalized Robinson decomposition; and (ii) RKHS regression with a product kernel $\mathcal{K}_g \odot \mathcal{K}_h$ (`ProdKernel`). We split the synthetic dataset into a training set and test set, fit different methods on the training data, and evaluate the fitted results on the test dataset; we repeat this procedure across multiple ranks.

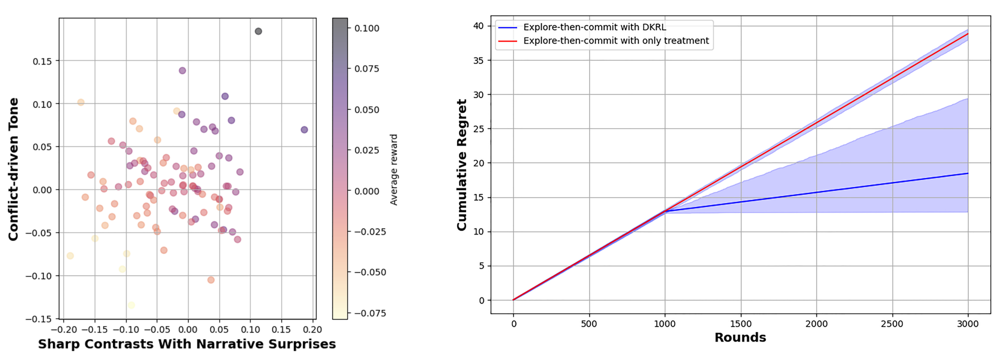
From Table 1, we can see that in these cases, DKRL achieves the best training and testing error. Neural network-based approaches are likely inferior here because they may not learn the low-rank representation; moreover, it is quite challenging to construct the relevant propensity functions for high-dimensional treatments (although this can be avoided in experimental settings). The low computational cost (measured on a Macbook Pro with a M2 Max chip) of kernel-based methods also enables more rapid iteration for large-scale A/B testing.

[2]The data is available at `https://huggingface.co/datasets/TrainingDataPro/asos-e-commerce-dataset`

Table 1: Upworthy Metrics by Rank and Method (Mean (Std), rounded to four decimals)

Rank	Method	Train Error	Test Error	Time (sec)
2	`SIN`	0.0105 (0.0006)	0.0104 (0.0017)	8.4686 (0.6584)
	`DKRL`	**0.0019 (0.0004)**	**0.0036 (0.0010)**	0.1907 (0.0339)
	`ProdKernel`	0.0043 (0.0002)	0.0080 (0.0013)	0.0014 (0.0002)
3	`SIN`	0.0113 (0.0006)	0.0113 (0.0016)	7.8103 (0.7073)
	`DKRL`	**0.0022 (0.0002)**	**0.0048 (0.0013)**	0.5062 (0.1448)
	`ProdKernel`	0.0048 (0.0002)	0.0090 (0.0013)	0.0014 (0.0002)
5	`SIN`	0.0143 (0.0006)	0.0144 (0.0018)	8.5115 (0.5318)
	`DKRL`	**0.0025 (0.0001)**	**0.0078 (0.0012)**	1.6933 (0.3752)
	`ProdKernel`	0.0062 (0.0002)	0.0115 (0.0014)	0.0015 (0.0003)
7	`SIN`	0.0171 (0.0007)	0.0172 (0.0021)	7.8928 (0.3724)
	`DKRL`	**0.0025 (0.0001)**	**0.0108 (0.0015)**	3.6143 (0.5942)
	`ProdKernel`	0.0072 (0.0002)	0.0136 (0.0017)	0.0015 (0.0005)

Dimension reduction and interpretation. Double kernel representation learning also enables low-dimensional summaries of high-dimensional features. Figure 1(a) presents the kernel representation for the treatment learned from the data. It shows that the components have a semantic interpretation (based on an evaluation by GPT 4o). More concretely, the first learned feature captures the scale of sharp contrasts with narrative surprises; the second captures the extent of a conflict-driven tone. Hence, we can view the values as semantic scores that can directly drive the average reward of the headlines and improve overall interpretability. In Table 2 of the Appendix, we show several example headlines with extreme semantic scores, showcasing how the features capture semantic meaning.



(a) (b)

Figure 1: Results on the Upworthy experiment. Panel (a) shows kernel representation of treatments in the semi-synthetic Upworthy Dataset. Panel (b) shows the cumulative regret in the evaluation of the bandit algorithm. The shaded area is the 15%-85% quantile range of the regret over simulations.

Evaluation of the bandit algorithm. We also evaluate the Explore-then-commit algorithm for online experimentation (Algorithm 2). We add explore-then-commit with only treatment representations as an additional baseline online experimentation method for comparison. Figure 1(b) presents the cumulative regret of Algorithm 2. As demonstrated by the theoretical results in Theorem 5.1, Algorithm 2 produces a sublinear regret in the long run; ignoring user-level covariates leads to failed exploration and thus linear regret. This highlights the importance of incorporating treatment-user interactions into both the outcome modeling and adaptive design strategy.

6.2 MIND dataset

Finally, we present results on the MIND dataset, an observational dataset with news recommended to users with unknown probabilities. We consider a semi-synthetic setup: suppose our goal is to train this recommendation system from scratch by adaptively collecting recommendation-click data to improve recommendation quality. We take x to be user features such as news category preference and historical news click embeddings, and z to be the embeddings for a set of candidate news to be recommended to users. All embeddings are taken from the knowledge graph embeddings that were originally included in the dataset. We consider a synthetic outcome model for the click-through rate: $y = z^\top \boldsymbol{\Theta}^\star x + \epsilon$, where $\boldsymbol{\Theta}^\star$ is a matrix that encodes the interaction mechanism between users and treatments; we can interpret this as a match between user preferences and news information. We consider $\boldsymbol{\Theta}^\star = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{V}^\top$, with $\boldsymbol{\Lambda}_i = \mathrm{Diag}\{i^{-q}\}$, and vary q to ensure that the interaction model varies from low rank (high eigenvalue decay rates) to high rank (low eigenvalue decay rates) setting. We compare our method with four baselines: LASSO, XG-boost, a feed-forward neural network, and a kernel regression. We finally combine the methods with different estimation strategies: "Z" only incorporates treatment information, "X" only includes covariate information, and "ZX" includes a concatenated (Z, X) vector.

Table 3 in the Appendix shows test RMSE. Overall, the performance of all methods is better with low-rank signals (larger q) than high-rank signals. However, DKRL exploits this low-rank structure most effectively, with the strongest performance. Even with a high-rank interaction matrix (e.g., a full-rank matrix), DKRL effectively exploits the similarity structure in the embeddings, improving accuracy and achieving comparable performance to XG Boost with combined features.

7 Conclusion

Summary. As content creation at scale becomes more readily available, developing new causal inference methods becomes increasingly important. In this work, we explore the design and analysis of GenAI-powered digital experimentation. We propose to use GenAI for efficient and novel treatment generation and to learn the corresponding heterogeneous treatment effect function using a double-kernel representation learning framework. We then propose an adaptive data collection algorithm for online experimentation. Finally, we provide theoretical guarantees and multiple numerical demonstrations.

Limitations and future directions. Several limitations remain. First, it is important to consider more complex experimentation settings that are important in practice, such as delayed responses (Shi et al., 2023, 2024b) or interference between participants (Li and Wager, 2022). Second, we established theoretical guarantees for the double-kernel representation learning framework in a fixed-basis setting. It would be interesting to generalize this theory to infinite-dimensional bases. Third, GenAI-powered experimentation raises key questions on issues like privacy and fairness, which are important to address independent of this current research. Fourth, it is natural to use other methods such as language model finetuning (Hu et al., 2022) or contrast learning (Zha et al., 2023) to learn improved embeddings and to generally improve the pipeline.

Acknowledgments and Disclosure of Funding

We thank the AC and four reviewers for constructive comments and feedback.

References

Abernethy, J., Bach, F., Evgeniou, T., and Vert, J.-P. (2006). Low-rank matrix factorization with attributes. *arXiv preprint cs/0611124*.

Athey, S., Bayati, M., Doudchenko, N., Imbens, G., and Khosravi, K. (2021). Matrix completion methods for causal panel data models. *Journal of the American Statistical Association*, 116(536):1716–1730.

Baek, T. H. (2023). Digital advertising in the age of generative ai. *Journal of Current Issues & Research in Advertising*, 44(3):249–251.

Bickel, P., Ritov, Y., and Tsybakov, A. (2008). Simultaneous analysis of lasso and dantzig selector. *The Annals of Statistics*, 37.

Bonilla, E. V., Chai, K., and Williams, C. (2007). Multi-task gaussian process prediction. *Advances in neural information processing systems*, 20.

Candes, E. J. and Plan, Y. (2011). Tight oracle inequalities for low-rank matrix recovery from a minimal number of noisy random measurements. *IEEE Transactions on Information Theory*, 57(4):2342–2359.

Chernozhukov, V., Chetverikov, D., Demirer, M., Duflo, E., Hansen, C., Newey, W., and Robins, J. (2018). Double/debiased machine learning for treatment and structural parameters.

Chi, Y., Lu, Y. M., and Chen, Y. (2019). Nonconvex optimization meets low-rank matrix factorization: An overview. *IEEE Transactions on Signal Processing*, 67(20):5239–5269.

Chowdhury, S. R. and Gopalan, A. (2017). On kernelized multi-armed bandits. In *International Conference on Machine Learning*, pages 844–853. PMLR.

Cui, X., Shi, L., Zhong, W., and Zou, C. (2023). Robust high-dimensional low-rank matrix estimation: Optimal rate and data-adaptive tuning. *Journal of Machine Learning Research*, 24(350):1–57.

Dimmery, D., Bakshy, E., and Sekhon, J. (2019). Shrinkage estimators in online experiments. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2914–2922.

Egami, N., Fong, C. J., Grimmer, J., Roberts, M. E., and Stewart, B. M. (2022). How to make causal inferences using texts. *Science Advances*, 8(42):eabg2652.

Esser, P., Fleissner, M., and Ghoshdastidar, D. (2024). Non-parametric representation learning with kernels. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 11910–11918.

Feder, A., Keith, K. A., Manzoor, E., Pryzant, R., Sridhar, D., Wood-Doughty, Z., Eisenstein, J., Grimmer, J., Reichart, R., Roberts, M. E., et al. (2022). Causal inference in natural language processing: Estimation, prediction, interpretation and beyond. *Transactions of the Association for Computational Linguistics*, 10:1138–1158.

Filippi, S., Cappe, O., Garivier, A., and Szepesvári, C. (2010). Parametric bandits: The generalized linear case. *Advances in neural information processing systems*, 23.

Fithian, W. and Mazumder, R. (2013). Flexible low-rank statistical modeling with side information. *arXiv preprint arXiv:1308.4211*.

Fong, C. and Grimmer, J. (2023). Causal inference with latent treatments. *American Journal of Political Science*, 67(2):374–389.

Frazier, P. I. (2018). A tutorial on bayesian optimization. *arXiv preprint arXiv:1807.02811*.

Gerber, A. S., Huber, G. A., Fang, A. H., and Reardon, C. E. (2018). The comparative effectiveness on turnout of positively versus negatively framed descriptive norms in mobilization campaigns. *American Politics Research*, 46(6):996–1011.

Gołąb-Andrzejak, E. (2023). The impact of generative ai and chatgpt on creating digital advertising campaigns. *Cybernetics and Systems*, pages 1–15.

Green, J., Druckman, J. N., Baum, M. A., Lazer, D., Ognyanova, K., Simonson, M. D., Lin, J., Santillana, M., and Perlis, R. H. (2023). Using general messages to persuade on a politicized scientific issue. *British Journal of Political Science*, 53(2):698–706.

Grimmer, M. and Woolley, M. (2014). Green marketing messages and consumers' purchase intentions: Promoting personal versus environmental benefits. *Journal of Marketing Communications*, 20(4):231–250.

Hao, B., Lattimore, T., and Wang, M. (2020). High-dimensional sparse linear bandits. *Advances in Neural Information Processing Systems*, 33:10753–10763.

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W., et al. (2022). Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3.

Ignatiadis, N. and Wager, S. (2019). Covariate-powered empirical bayes estimation. *Advances in Neural Information Processing Systems*, 32.

Imai, K. and Nakamura, K. (2024). Causal representation learning with generative artificial intelligence: Application to texts as treatments. *arXiv preprint arXiv:2410.00903*.

Jain, P., Netrapalli, P., and Sanghavi, S. (2013). Low-rank matrix completion using alternating minimization. In *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, pages 665–674.

Jin, H., Ma, Y., and Jiang, F. (2022). Matrix completion with covariate information and informative missingness. *Journal of Machine Learning Research*, 23(180):1–62.

Joachims, T. (1998). Text categorization with support vector machines: Learning with many relevant features. In *European conference on machine learning*, pages 137–142. Springer.

Jun, K.-S., Willett, R., Wright, S., and Nowak, R. (2019). Bilinear bandits with low-rank structure. In *International Conference on Machine Learning*, pages 3163–3172. PMLR.

Kaddour, J., Zhu, Y., Liu, Q., Kusner, M. J., and Silva, R. (2021). Causal effect inference for structured treatments. In Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W., editors, *Advances in Neural Information Processing Systems*, volume 34, pages 24841–24854. Curran Associates, Inc.

Klopp, O. (2014). Noisy low-rank matrix completion with general sampling distribution. *Bernoulli*, 20(1):282–303.

Komiyama, J. and Imaizumi, M. (2024). High-dimensional contextual bandit problem without sparsity. *Advances in Neural Information Processing Systems*, 36.

Kumar, M. and Kapoor, A. (2024). Generative ai and personalized video advertisements. *Available at SSRN 4614118*.

Leung, X. Y., Bai, B., and Erdem, M. (2017). Hotel social media marketing: a study on message strategy and its effectiveness. *Journal of Hospitality and Tourism Technology*, 8(2):239–255.

Li, L., Lu, Y., and Zhou, D. (2017). Provably optimal algorithms for generalized linear contextual bandits. In *International Conference on Machine Learning*, pages 2071–2080. PMLR.

Li, S. and Wager, S. (2022). Network interference in micro-randomized trials. *arXiv preprint arXiv:2202.05356*.

Liu, Q., Chen, Z., and Wong, W. H. (2024). An encoding generative modeling approach to dimension reduction and covariate adjustment in causal inference with observational studies. *Proceedings of the National Academy of Sciences*, 121(23):e2322376121.

Lu, T., Pál, D., and Pál, M. (2010). Contextual multi-armed bandits. In *Proceedings of the Thirteenth international conference on Artificial Intelligence and Statistics*, pages 485–492. JMLR Workshop and Conference Proceedings.

Lu, X., Shi, L., Liu, H., and Ding, P. (2025). Conditional cross-fitting for unbiased machine-learning-assisted covariate adjustment in randomized experiments. *arXiv preprint arXiv:2508.15664*.

Lu, Y., Meisami, A., and Tewari, A. (2021). Low-rank generalized linear bandit problems. In *International Conference on Artificial Intelligence and Statistics*, pages 460–468. PMLR.

Ma, C., Pathak, R., and Wainwright, M. J. (2023). Optimally tackling covariate shift in rkhs-based nonparametric regression. *The Annals of Statistics*, 51(2):738–761.

Mao, X., Chen, S. X., and Wong, R. K. (2019). Matrix completion with covariate information. *Journal of the American Statistical Association*, 114(525):198–210.

Martinez-Cantin, R. (2014). Bayesopt: a bayesian optimization library for nonlinear optimization, experimental design and bandits. *J. Mach. Learn. Res.*, 15(1):3735–3739.

Matias, J. N., Munger, K., Le Quere, M. A., and Ebersole, C. (2021). The upworthy research archive, a time series of 32,487 experiments in us media. *Scientific Data*, 8(1):195.

Mendelson, S. and Neeman, J. (2010). Regularization in kernel learning. *The Annals of Statistics*, 38(1):526 – 565.

Negahban, S. and Wainwright, M. J. (2012). Restricted strong convexity and weighted matrix completion: Optimal bounds with noise. *The Journal of Machine Learning Research*, 13(1):1665–1697.

Nie, X. and Wager, S. (2021). Quasi-oracle estimation of heterogeneous treatment effects. *Biometrika*, 108(2):299–319.

Oh, M.-h., Iyengar, G., and Zeevi, A. (2021). Sparsity-agnostic lasso bandit. In *International Conference on Machine Learning*, pages 8271–8280. PMLR.

Recht, B., Fazel, M., and Parrilo, P. A. (2010). Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM review*, 52(3):471–501.

Robinson, P. M. (1988). Root-n-consistent semiparametric regression. *Econometrica: Journal of the Econometric Society*, pages 931–954.

Rohde, A. and Tsybakov, A. B. (2011). Estimation of high-dimensional low-rank matrices. *Annals of Statistics*, 39(2):887–930.

Rusmevichientong, P. and Tsitsiklis, J. N. (2010). Linearly parameterized bandits. *Mathematics of Operations Research*, 35(2):395–411.

Salovey, P. and Williams-Piehota, P. (2004). Field experiments in social psychology: Message framing and the promotion of health protective behaviors. *American Behavioral Scientist*, 47(5):488–505.

Schölkopf, B., Smola, A., and Müller, K.-R. (1997). Kernel principal component analysis. In *International conference on artificial neural networks*, pages 583–588. Springer.

Shi, L., Wang, G., and Zou, C. (2024a). Low-rank matrix estimation in the presence of change-points. *Journal of Machine Learning Research*, 25(220):1–71.

Shi, L., Wang, J., and Wu, T. (2023). Statistical inference on multi-armed bandits with delayed feedback. In *International Conference on Machine Learning*, pages 31328–31352. PMLR.

Shi, L., Wei, W., and Wang, J. (2024b). Using surrogates in covariate-adjusted response-adaptive randomization experiments with delayed outcomes. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Shi, L. and Zou, C. (2020). Noisy low-rank matrix completion under general bases. *Stat*, 9(1):e303.

Singh, R., Xu, L., and Gretton, A. (2024). Kernel methods for causal functions: dose, heterogeneous and incremental response curves. *Biometrika*, 111(2):497–516.

Srinivas, N., Krause, A., Kakade, S. M., and Seeger, M. (2009). Gaussian process optimization in the bandit setting: No regret and experimental design. *arXiv preprint arXiv:0912.3995*.

Sun, T., Viswanathan, S., and Zheleva, E. (2021). Creating social contagion through firm-mediated message design: Evidence from a randomized field experiment. *Management Science*, 67(2):808–827.

Tierney, G., Katta, S., Bail, C., Hillygus, S., and Volfovsky, A. (2025). A design-based solution for causal inference with text: Can a language model be too large? *arXiv preprint arXiv:2510.08758*.

Vakili, S., Ahmed, D., Bernacchia, A., and Pike-Burke, C. (2023). Delayed feedback in kernel bandits. In *International Conference on Machine Learning*, pages 34779–34792. PMLR.

Valko, M., Korda, N., Munos, R., Flaounas, I., and Cristianini, N. (2013). Finite-time analysis of kernelised contextual bandits. *arXiv preprint arXiv:1309.6869*.

Wainwright, M. J. (2019). *High-dimensional statistics: A non-asymptotic viewpoint*, volume 48. Cambridge university press.

Wang, W., Wei, F., Dong, L., Bao, H., Yang, N., and Zhou, M. (2020). Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *Advances in Neural Information Processing Systems*, 33:5776–5788.

Williams, C. and Seeger, M. (2000). Using the nyström method to speed up kernel machines. *Advances in neural information processing systems*, 13.

Williams, C. K. and Rasmussen, C. E. (2006). *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA.

Wu, F., Qiao, Y., Chen, J.-H., Wu, C., Qi, T., Lian, J., Liu, D., Xie, X., Gao, J., Wu, W., et al. (2020). Mind: A large-scale dataset for news recommendation. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 3597–3606.

Zha, K., Cao, P., Son, J., Yang, Y., and Katabi, D. (2023). Rank-n-contrast: learning continuous representations for regression. *Advances in Neural Information Processing Systems*, 36:17882–17903.

Zhou, T., Shan, H., Banerjee, A., and Sapiro, G. (2012). Kernelized probabilistic matrix factorization: Exploiting graphs and side information. In *Proceedings of the 2012 SIAM international Conference on Data mining*, pages 403–414. SIAM.

A Kernel Methods

Kernel-based methods have been widely used in related practices such as representation learning (Esser et al., 2024), dimension reduction (Schölkopf et al., 1997), natural language processing Joachims (1998), among others. We do a brief overview of RKHS methods. Consider a positive definite kernel $\mathcal{K} : \mathcal{Z} \times \mathcal{Z} \to \mathbb{R}$ supported on a compact d-dimensional set $\mathcal{Z} \subset \mathbb{R}^d$. A Hilbert space $\mathbb{H}_{\mathcal{K}}$ of functions on $\mathcal{Z}$ equipped with an inner product $\langle \cdot \, , \, \cdot \rangle_{\mathbb{H}_{\mathcal{K}}}$ is called a reproducing kernel Hilbert space (RKHS) with reproducing kernel K if the following properties are satisfied:

1. for all $z \in \mathcal{Z}$, $\mathcal{K}(\cdot, z) \in \mathbb{H}_{\mathcal{K}}$;
2. for all $z \in \mathcal{Z}$ and $f \in \mathbb{H}_{\mathcal{K}}$, $\langle f \, , \, \mathcal{K}(\cdot, z) \rangle_{\mathbb{H}_{\mathcal{K}}} = f(z)$ (reproducing property).

A natural idea for estimation and policy learning is to incorporate embedding information with RKHS methods. Given sample (t_i, y_i), we can fit the following penalized nonlinear least squares:

$$\widehat{f} = \operatorname*{arg\,min}_{f \in \mathbb{H}_{\mathcal{K}}} \frac{1}{2n} \sum_{i=1}^{n} (y_i - f(z_i))^2 + \lambda_n \|f\|_{\mathbb{H}_{\mathcal{K}}}.$$

Then we can predict the estimated outcome response given a specific treatment $z \in \mathcal{Z}$ as $\widehat{f}(z)$. By the representer theorem, this leads to a program based on the data:

$$\widehat{\alpha} = \operatorname*{arg\,min}_{\alpha \in \mathbb{R}^n} \frac{1}{2n} \sum_{i=1}^{n} \{y_i - \sum_{i=1}^{n} \alpha_j \mathcal{K}(z_i, z_j)\}^2 + \lambda_n \alpha^\top \mathcal{K} \alpha,$$

which gives the interpolator

$$\widehat{f}(z) = \sum_{i=1}^{n} \widehat{\alpha}_j \mathcal{K}(z, z_j).$$

Another viewpoint of the RKHS-based method is that we can put a distribution on the functions and fit a Gaussian process model (Williams and Rasmussen, 2006):

$$y \sim \mathcal{N}(0, \sigma^2), \quad f \sim \mathrm{GP}(0, \mathcal{K}). \tag{12}$$

Then the interpolator is equivalent to a posterior prediction from (12).

In particular, if the kernel $\mathcal{K}$ is a linear inner product of finite dimensional features $\phi(z)$, the above program is also equivalent to ridge regression:

$$\widehat{f}(z) = \phi(z)^\top \widehat{\beta},$$

where

$$\widehat{\beta} = \frac{1}{2n} \sum_{i=1}^{n} \{y_i - \phi(z_i)^\top \beta\}^2 + \lambda_n \|\beta\|_2^2.$$

Such connection has been pointed out by previous literature, e.g. Valko et al. (2013).

This is also equivalent to a Bayesian linear model:

$$y \sim \mathcal{N}(\phi(z)^\top \beta, \sigma^2 I), \text{ where } \beta \sim \mathcal{N}(0, \sigma_0^2 I).$$

B Connection with classical approaches

For intuition we now draw connections between the proposed method to two classical approaches: kernelized probability matrix factorization, which is introduced by Zhou et al. (2012) and serves as the Gaussian process counterpart of our method, and low-rank matrix factorization algorithms.

Connection with kernelized probability matrix factorization. The proposed method is closely related to kernelized probability matrix factorization (Zhou et al., 2012). To see this, we revisit the program (8) but instead of using the parametrization $\boldsymbol{U}$ and $\boldsymbol{V}$ consider the following transformation:

$$\boldsymbol{R} = \mathcal{K}_g\boldsymbol{U} \in \mathbb{R}^{n\times r}, \quad \boldsymbol{S} = \mathcal{K}_h\boldsymbol{V} \in \mathbb{R}^{n\times r}.$$

Here, $\boldsymbol{R}$ and $\boldsymbol{S}$ are the intrinsic representations of the treatment and covariate-level information, respectively. Then (8) can be written as

$$(\widehat{\boldsymbol{R}}, \widehat{\boldsymbol{S}}) = \arg\min_{\boldsymbol{R},\boldsymbol{S}} \frac{1}{2n}\sum_{k=1}^{n}\left(y_k - \boldsymbol{R}_k^{\mathrm{R}\top}\boldsymbol{S}_k^{\mathrm{R}}\right)^2 + \lambda_n\sum_{l=1}^{r}\left(\boldsymbol{R}_l^{\mathrm{C}\top}\mathcal{K}_g^{-1}\boldsymbol{R}_l^{\mathrm{C}} + \boldsymbol{S}_l^{\mathrm{C}\top}\mathcal{K}_h^{-1}\boldsymbol{S}_l^{\mathrm{C}}\right).$$

We can instead follow the kernelized probabilistic factorization in Zhou et al. (2012). Under the following data-generating process:

$$\boldsymbol{R}_l^{\mathrm{C}} \sim \mathrm{GP}(0, \mathcal{K}_g), \quad \boldsymbol{S}_l^{\mathrm{C}} \sim \mathrm{GP}(0, \mathcal{K}_h), \quad l \in [r],$$

and

$$y_k \sim \mathcal{N}(\boldsymbol{R}_k^{\mathrm{R}\top}\boldsymbol{S}_k^{\mathrm{R}}, \sigma^2),$$

Zhou et al. (2012) showed that the log-posterior over the latent features $\boldsymbol{R}$ and $\boldsymbol{S}$ is given by

$$\begin{aligned}\log p(\boldsymbol{R}, \boldsymbol{S} \mid \boldsymbol{y}, \sigma^2) = &-\frac{1}{2\sigma^2}\sum_{k=1}^{n}(y_k - \boldsymbol{R}_k^{\mathrm{R}\top}\boldsymbol{S}_k^{\mathrm{R}})^2 - \frac{1}{2}\sum_{l=1}^{r}\left(\boldsymbol{R}_l^{\mathrm{C}\top}\mathcal{K}_g^{-1}\boldsymbol{R}_l^{\mathrm{C}} + \boldsymbol{S}_l^{\mathrm{C}\top}\mathcal{K}_h^{-1}\boldsymbol{S}_l^{\mathrm{C}}\right)\\ &- n\log\sigma^2 - \frac{r}{2}\{\log(|\mathcal{K}_g|) + \log(|\mathcal{K}_h|)\} + C,\end{aligned} \tag{13}$$

where C is some universal constant that does not depend on $\boldsymbol{R}$ and $\boldsymbol{S}$. Maximizing the log-posterior (13) is equivalent to minimizing

$$L(\boldsymbol{R}, \boldsymbol{S}) = \frac{1}{2\sigma^2}\sum_{k=1}^{n}(y_k - \boldsymbol{R}_k^{\mathrm{R}\top}\boldsymbol{S}_k^{\mathrm{R}})^2 + \frac{1}{2}\sum_{l=1}^{r}\left(\boldsymbol{R}_l^{\mathrm{C}\top}\mathcal{K}_g^{-1}\boldsymbol{R}_l^{\mathrm{C}} + \boldsymbol{S}_l^{\mathrm{C}\top}\mathcal{K}_h^{-1}\boldsymbol{S}_l^{\mathrm{C}}\right),$$

which is equivalent to (8).

Connection with low-rank matrix learning with finite-dimensional features. Consider the kernel given by the inner product of some feature mapping $\phi \in \mathbb{R}^p$:

$$\mathcal{K}_g(z, z') = \phi(z)^\top\phi(z'),$$

with feature matrix $\Phi = [\phi(z_1), \ldots, \phi(z_n)]^\top \in \mathbb{R}^{n\times p}$. Then the kernel matrix has the expression:

$$\mathcal{K}_g = \Phi\Phi^\top.$$

We prove the following reformulation results:

Proposition B.1. *Suppose the feature matrix Φ has full (column or row) rank. Then*

$$(\widehat{\boldsymbol{U}}^\star, \widehat{\boldsymbol{V}}^\star) = \arg\min_{\boldsymbol{U},\boldsymbol{V}} \frac{1}{2n}\sum_{k=1}^{n}\{y_k - \phi(z_k)^\top\boldsymbol{U}^\star\boldsymbol{V}^{\star\top}\psi(x_k)\}^2 + \lambda_n(\|\boldsymbol{U}^\star\|_F^2 + \|\boldsymbol{V}^\star\|_F^2). \tag{14}$$

Now following Proposition B.1, we have the following equivalent formulation based on nuclear-norm penalization:

Proposition B.2. *The program* (14) *is equivalent to the following low-rank learning program:*

$$\widehat{\boldsymbol{\Theta}} = \arg\min_{\boldsymbol{\Theta}} \frac{1}{2n}\sum_{k=1}^{n}\{y_k - \phi(z_k)^\top\boldsymbol{\Theta}\psi(x_k)\}^2 + 2\lambda_n\|\boldsymbol{\Theta}\|_*. \tag{15}$$

This equivalence connects the matrix factorization approach with nuclear norm penalization for low-rank matrix estimation. The key insight is due to the variational expression of the nuclear norm (Recht et al., 2010):

$$\|\boldsymbol{\Theta}\|_* = \min_{\boldsymbol{U}^\star,\boldsymbol{V}^\star:\boldsymbol{\Theta}=\boldsymbol{U}^\star\boldsymbol{V}^{\star\top}} \frac{1}{2}(\|\boldsymbol{U}\|_F^2 + \|\boldsymbol{V}\|_F^2).$$

C Additional results

C.1 Bounds on $\boldsymbol{\Theta}$

The rate in Theorem 4.2 also implies a rate for $\boldsymbol{\Theta}$ when the number of tested variants and candidate users is large in the sense that $d_1 > p$ and $d_2 > q$. To see this, we can express $\boldsymbol{\Theta}^\star$ with $\boldsymbol{\Gamma}^\star$, as defined in (11):

$$\boldsymbol{\Theta}^\star = (\boldsymbol{Z}\boldsymbol{Z}^\top)^{-1}\boldsymbol{Z}\boldsymbol{\Gamma}^\star\boldsymbol{X}^\top(\boldsymbol{X}\boldsymbol{X}^\top)^{-1}.$$

This motivates an estimator:

$$\widehat{\boldsymbol{\Theta}} = (\boldsymbol{Z}\boldsymbol{Z}^\top)^{-1}\boldsymbol{Z}\widehat{\boldsymbol{\Gamma}}\boldsymbol{X}^\top(\boldsymbol{X}\boldsymbol{X}^\top)^{-1}. \tag{16}$$

Then we have the following bound:

$$\begin{aligned}\|\widehat{\boldsymbol{\Theta}} - \boldsymbol{\Theta}^\star\|_F^2 \leq & \rho_{\max}\{(\boldsymbol{Z}\boldsymbol{Z}^\top)^{-1}\}\rho_{\max}\{(\boldsymbol{X}\boldsymbol{X}^\top)^{-1}\}\|\widehat{\boldsymbol{\Gamma}} - \boldsymbol{\Gamma}^\star\|_F^2 \\ \leq & \rho_{\min}\{(\boldsymbol{Z}\boldsymbol{Z}^\top/d_1)\}^{-1}\rho_{\min}\{(\boldsymbol{X}\boldsymbol{X}^\top/d_2)\}^{-1}\cdot\lambda_n^2 r.\end{aligned}$$

Therefore, as long as the minimal eigenvalue of $\boldsymbol{Z}\boldsymbol{Z}^\top$ and $\boldsymbol{X}\boldsymbol{X}^\top$ are not degenerate, the estimator (16) is also consistent.

D Technical proofs

D.1 Proof of Theorem 4.1

Proof of Theorem 4.1. The RKHS space $\mathbb{H}_g$ and $\mathbb{H}_h$ has the following decomposition:

$$\mathbb{H}_g = \mathbb{H}_g^S \oplus \mathbb{H}_g^\perp, \quad \mathbb{H}_h = \mathbb{H}_h^S \oplus \mathbb{H}_h^\perp,$$

where $\mathbb{H}_g^S$ and $\mathbb{H}_h^S$ are the orthogonal projection into the sample-generated function subspace. For any $\boldsymbol{g}$ and $\boldsymbol{h}$, we have the decomposition

$$\boldsymbol{g} = \boldsymbol{g}^S + \boldsymbol{g}^\perp, \quad \boldsymbol{h} = \boldsymbol{h}^S + \boldsymbol{h}^\perp.$$

Then for any i, $\boldsymbol{g}(z_i) = \boldsymbol{g}^S(z_i) + \boldsymbol{g}^\perp(z_i) = \boldsymbol{g}^S(z_i)$, $\boldsymbol{h}(x_i) = \boldsymbol{h}^S(x_i) + \boldsymbol{h}^\perp(x_i) = \boldsymbol{h}^S(x_i)$. Meanwhile, due to the projection, we have a reduction in the RKHS norm:

$$\|g_l^S\|_{\mathcal{K}_g}^2 \leq \|g_l\|_{\mathcal{K}_g}^2, \quad \|h_l^S\|_{\mathcal{K}_h}^2 \leq \|h_l\|_{\mathcal{K}_h}^2.$$

Therefore, by replacing $\boldsymbol{g}$ with $\boldsymbol{g}^S$ and also $\boldsymbol{h}$ with $\boldsymbol{h}^S$, the objective function never increases. Hence, the optimal solution must lie in $\mathbb{H}_{\mathcal{K}_g}^S$ and $\mathbb{H}_{\mathcal{K}_h}^S$. □

D.2 Proof of Proposition B.1

Proof of Proposition B.1. Case 1. When $n \geq p$ and the feature matrix $\Phi \in \mathbb{R}^{n\times p}$ has full column rank, we can conclude that the mapping $\boldsymbol{U} \mapsto \Phi^\top\boldsymbol{U}$ is a surjective mapping from $\mathbb{R}^{n\times r}$ to $\mathbb{R}^{p\times r}$. Therefore, (8) is equivalent to solving the feature-based optimization program:

$$(\widehat{\boldsymbol{U}}^\star, \widehat{\boldsymbol{V}}^\star) = \min_{\boldsymbol{U},\boldsymbol{V}} \frac{1}{2n}\sum_{k=1}^n \{y_k - \phi(z_k)^\top\boldsymbol{U}^\star\boldsymbol{V}^{\star\top}\psi(x_k)\}^2 + \lambda_n(\|\boldsymbol{U}^\star\|_F^2 + \|\boldsymbol{V}^\star\|_F^2).$$

Case 2. When $n < p$ and the feature matrix $\Phi \in \mathbb{R}^{n\times p}$ has full row rank, we start from the program (14). For any $\boldsymbol{U}^\star$, there exists $\boldsymbol{U}$, such that

$$\Phi\Phi^\top\boldsymbol{U} = \Phi\boldsymbol{U}^\star,$$

as one can take $\boldsymbol{U} = (\Phi\Phi^\top)^{-1}\Phi\boldsymbol{U}^\star$. Therefore, we can always reparametrize (14) as (8).

□

D.3 Proof of Proposition B.2

Proof of Proposition B.2. The nuclear norm $\|\mathbf{\Theta}\|_*$ of a matrix has the following variational representation:

$$\|\mathbf{\Theta}\|_* = \min_{\boldsymbol{U}\boldsymbol{V}^\top = \mathbf{\Theta}} \frac{1}{2}(\|\boldsymbol{U}\|_F^2 + \|\boldsymbol{V}\|_F^2).$$

Using this result, we can deduce the equivalence between the program (14) and (15):

$$\begin{aligned}
&\min_{\mathbf{\Theta}} p \frac{1}{2n}\sum_{k=1}^{n}\{y_k - \phi(z_k)^\top \mathbf{\Theta}\psi(x_k)\}^2 + 2\lambda_n\|\mathbf{\Theta}\|_* \\
=&\min_{\mathbf{\Theta}} \frac{1}{2n}\sum_{k=1}^{n}\{y_k - \phi(z_k)^\top \mathbf{\Theta}\psi(x_k)\}^2 + 2\lambda_n \left\{ \min_{\boldsymbol{U}^\star, \boldsymbol{V}^\star : \boldsymbol{U}^\star \boldsymbol{V}^{\star\top} = \mathbf{\Theta}} \frac{1}{2}(\|\boldsymbol{U}^\star\|_F^2 + \|\boldsymbol{V}^\star\|_F^2) \right\} \\
=&\min_{\mathbf{\Theta}} \min_{\boldsymbol{U}^\star, \boldsymbol{V}^\star : \boldsymbol{U}^\star \boldsymbol{V}^{\star\top} = \mathbf{\Theta}} \frac{1}{2n}\sum_{k=1}^{n}\{y_k - \phi(z_k)^\top \boldsymbol{U}^\star \boldsymbol{V}^{\star\top}\psi(x_k)\}^2 + \lambda_n(\|\boldsymbol{U}^\star\|_F^2 + \|\boldsymbol{V}^\star\|_F^2) \\
=&\min_{\boldsymbol{U}^\star, \boldsymbol{V}^\star} \frac{1}{2n}\sum_{k=1}^{n}\{y_k - \phi(z_k)^\top \boldsymbol{U}^\star \boldsymbol{V}^{\star\top}\psi(x_k)\}^2 + \lambda_n(\|\boldsymbol{U}^\star\|_F^2 + \|\boldsymbol{V}^\star\|_F^2).
\end{aligned}$$

□

D.4 Statement and proof of a more general version of Theorem 4.2

In this section, we prove a more general version of Theorem 4.2, which extends the results to an approximate low-rank setting. More concretely, let $\rho_k(\mathbf{\Gamma})$ be the k-th largest singular value of matrix $\mathbf{\Gamma}$. We define the following notion of ℓ_q-ball in the matrix space:

Definition D.1 (ℓ_q ball)**.** The ℓ_q ball with radius R_q is defined as the following set:

$$\mathbb{B}_q(R_q) = \left\{ \mathbf{\Gamma} \in \mathbb{R}^{d_1 \times d_2} : \sum_{k=1}^{\min\{d_1, d_2\}} \rho_k(\mathbf{\Gamma})^q \le R_q \right\}, \text{ for } q \in [0, 1).$$

When $q = 0$, this definition becomes the exact low-rank condition. When $q > 0$, the definition allows the existence of many small positive singular values, decaying at a proper rate (Negahban and Wainwright, 2012; Rohde and Tsybakov, 2011; Shi et al., 2024a; Shi and Zou, 2020; Cui et al., 2023).

Theorem D.2. *Assume that the noise ϵ_i's are i.i.d. and sub-exponential, and the true signal matrix $\mathbf{\Gamma}^\star$ belongs to the ℓ_q ball $\mathbb{B}_q(R_q)$ and entry bounded by a. With probability greater than $1 - c_1' \exp(-c_2' d \log d)$, the estimator $\widehat{\mathbf{\Gamma}}$ satisfies*

$$\frac{1}{d_1 d_2}\|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}^\star\|_F^2 \le C R_q \lambda_n^{2-q}, \quad \textit{where} \quad \lambda_n \asymp \max\left\{ \frac{1}{\sqrt{n}}\|\widehat{m}(x) - m(x)\|_2, \sqrt{\frac{d \log d}{n}} \right\}.$$

Proof of Theorem 4.2. For the convenience of analysis, we introduce the sampling operator $\mathfrak{S} : \mathbb{R}^{d_1 \times d_2} \to \mathbb{R}^{n \times 1}$:

$$\mathfrak{S}(\mathbf{\Gamma}) = (e_{z,1}^\top \mathbf{\Gamma} e_{x,1}, \ldots, e_{z,n}^\top \mathbf{\Gamma} e_{x,n})^\top.$$

$\mathfrak{S}$ is a linear operator. The adjoint operator, $\mathfrak{S}^\star : \mathbb{R}^{n \times 1} \to \mathbb{R}^{d_1 \times d_2}$, is defined as

$$\mathfrak{S}^\star(\boldsymbol{\epsilon}) = \sum_{i=1}^{n} \epsilon_i e_{x,i} e_{z,i}^\top.$$

The operator satisfies

$$\langle \mathfrak{S}(\mathbf{\Gamma}), \epsilon \rangle = \langle \mathbf{\Gamma}, \mathfrak{S}^\star(\epsilon) \rangle.$$

Step 1. Deriving a basic inequality. By the minimization program, for the estimated $\widehat{\mathbf{\Gamma}}$, we have

$$\frac{1}{2n}\|\boldsymbol{y}-\widehat{m}(\boldsymbol{x})-\mathfrak{S}(\widehat{\mathbf{\Gamma}})\|_2^2+\lambda_n\|\widehat{\mathbf{\Gamma}}\|_*\le\frac{1}{2n}\|\boldsymbol{y}-\widehat{m}(\boldsymbol{x})-\mathfrak{S}(\mathbf{\Gamma}^\star)\|_2^2+\lambda_n\|\mathbf{\Gamma}^\star\|_*.$$

This gives

$$\begin{aligned}
&\frac{1}{2n}\|\mathfrak{S}(\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star)\|_2^2\\
&\le\frac{1}{n}\left\langle\widehat{\epsilon},\,\mathfrak{S}(\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star)\right\rangle+\lambda_n\|\mathbf{\Gamma}^\star\|_*-\lambda_n\|\widehat{\mathbf{\Gamma}}\|_*\\
&=\frac{1}{n}\left\langle\widehat{m}(x)-m(x)\,,\,\mathfrak{S}(\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star)\right\rangle\\
&+\frac{1}{n}\left\langle\mathfrak{S}^\star(\epsilon)\,,\,\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star\right\rangle+\lambda_n\|\mathbf{\Gamma}^\star\|_*-\lambda_n\|\widehat{\mathbf{\Gamma}}\|_*\\
&\le\frac{1}{n}\|\widehat{m}(x)-m(x)\|_2\|\mathfrak{S}(\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star)\|_2+\frac{1}{n}\|\mathfrak{S}^\star(\epsilon)\|_{\mathrm{op}}\|\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star\|_*+\lambda_n\|\mathbf{\Gamma}^\star\|_*-\lambda_n\|\widehat{\mathbf{\Gamma}}\|_*.
\end{aligned}\tag{17}$$

Step 2. A cone inequality for the difference. Now we consider the projection of the difference $\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star$ onto a subspace that will lead to a decomposition into a low-rank part and a remainder that is relatively small in scale.

Let the singular value decomposition (SVD) of $\mathbf{\Gamma}^\star$ be $\mathbf{\Gamma}^\star=\boldsymbol{U}\mathbf{\Lambda}\boldsymbol{V}^\top$, where $\boldsymbol{U}\in\mathbb{R}^{d_1\times d_1}$ and $\boldsymbol{V}\in\mathbb{R}^{d_2\times d_2}$ are orthonormal matrices and $\mathbf{\Lambda}$ is the singular value matrix with descending diagonals. For any matrix $\mathbf{\Delta}\in\mathbb{R}^{d_1\times d_2}$, if we write

$$\boldsymbol{U}^\top\mathbf{\Delta}\boldsymbol{V}=\begin{pmatrix}\mathbf{\Omega}_{11}&\mathbf{\Omega}_{12}\\\mathbf{\Omega}_{21}&\mathbf{\Omega}_{22}\end{pmatrix},$$

with $\mathbf{\Omega}_{11}\in\mathbb{R}^{r\times r}$ and $\mathbf{\Omega}_{22}\in\mathbb{R}^{(d_1-r)\times(d_2-r)}$. We can define a projection

$$\Pi(\mathbf{\Delta})=\begin{pmatrix}\mathbf{\Omega}_{11}&\mathbf{\Omega}_{12}\\\mathbf{\Omega}_{21}&\mathbf{0}_{(d_1-r)\times(d_2-r)}\end{pmatrix},\quad\Pi^\perp(\mathbf{\Delta})=\begin{pmatrix}\mathbf{0}_{r\times r}&\mathbf{0}_{r\times(d_2-r)}\\\mathbf{0}_{(d_1-r)\times r}&\mathbf{\Omega}_{22}\end{pmatrix}.$$

With this decomposition, we always have $\mathrm{rank}(\Pi(\mathbf{\Delta}))\le 2r$. Now we choose an effective rank by trimming the matrix using a threshold, τ, and we will discuss how to pick τ later.

Since $\mathbf{\Gamma}^\star\in\mathbb{B}_q(R_q)$, we have

$$r\tau^q\le\sum_{k=1}^r\rho_k(\mathbf{\Delta})^q\le\sum_{k=1}^m\rho_k(\mathbf{\Delta})^q\le R_q,\quad r\le R_q\tau^{-q}.\tag{18}$$

Meanwhile, we also have

$$\|\Pi^\perp(\mathbf{\Delta})\|_*=\sum_{k=r+1}^m\rho_k(\mathbf{\Delta})\le\tau\sum_{k=r+1}^m\big(\frac{\rho_k(\mathbf{\Delta})}{\tau}\big)^q\le R_q\tau^{1-q}.\tag{19}$$

Using the triangle inequality, we have

$$\begin{aligned}
\|\widehat{\mathbf{\Gamma}}\|_*&=\|\mathbf{\Gamma}^\star+\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star\|_*\\
&=\|\mathbf{\Gamma}^\star+\Pi(\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star)+\Pi^\perp(\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star)\|_*\\
&\ge\|\Pi(\mathbf{\Gamma}^\star)+\Pi^\perp(\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star)\|_*-\|\Pi(\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star)\|_*-\|\Pi^\perp(\mathbf{\Gamma}^\star)\|_*\\
&=\|\Pi(\mathbf{\Gamma}^\star)\|_*+\|\Pi^\perp(\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star)\|_*-\|\Pi(\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star)\|_*-\|\Pi^\perp(\mathbf{\Gamma}^\star)\|_*,
\end{aligned}$$

which further gives

$$\|\mathbf{\Gamma}^\star\|_*-\|\widehat{\mathbf{\Gamma}}\|_*\le\|\Pi(\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star)\|_*-\|\Pi^\perp(\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star)\|_*+2\|\Pi^\perp(\mathbf{\Gamma}^\star)\|_*.$$

By choosing λ_n that satisfies

$$\lambda_n\ge\max\left\{\frac{1}{\sqrt{n}}\|\widehat{m}(x)-m(x)\|_2,\frac{2}{n}\|\mathfrak{S}^\star(\epsilon)\|_{\mathrm{op}}\right\},$$

together with (17), we have

$$
\begin{aligned}
&\frac{1}{2n}\|\mathfrak{S}(\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star)\|_2^2+\frac{\lambda_n}{2}\|\Pi^\perp(\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star)\|_*\\
\le&\frac{\lambda_n}{\sqrt{n}}\|\mathfrak{S}(\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star)\|_2+\frac{3\lambda_n}{2}\|\Pi(\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star)\|_*+2\lambda_n\|\Pi^\perp(\mathbf{\Gamma}^\star)\|_*. \qquad (20)
\end{aligned}
$$

For the term $2n^{-1}\|\mathfrak{S}^\star(\epsilon)\|_{\rm op}$, Negahban and Wainwright (2012) established the following bound in the proof of their Corollary 1:

$$
\mathbb{P}\left\{2n^{-1}\|\mathfrak{S}^\star(\epsilon)\|_{\rm op}\ge c_1\nu\sqrt{d\log d/n}\right\}\le c_2\exp(-c_3 d\log d).
$$

Step 3. A final bound. We will prove the following result:

Consider two cases.

Case 1. Suppose we have

$$
\frac{1}{4n}\|\mathfrak{S}(\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star)\|_2^2\ge\|\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star\|_F^2.
$$

Now using

$$
\|\Pi(\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star)\|_*\le\sqrt{2r}\|\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star\|_F
$$

and

$$
\frac{\lambda_n}{\sqrt{n}}\|\mathfrak{S}(\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star)\|_2\le\lambda_n^2+\frac{1}{4n}\|\mathfrak{S}(\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star)\|_2^2,\ \text{(Young's Inequality)}
$$

with (20), we have

$$
\|\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star\|_F^2\le\frac{1}{4n}\|\mathfrak{S}(\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star)\|_2^2\le\lambda_n^2+\frac{9\lambda_n^2 r}{4}+\frac{1}{2}\|\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star\|_F^2+2\lambda_n\|\Pi^\perp(\mathbf{\Gamma}^\star)\|_*.
$$

Using (18) and (19), the above gives

$$
\|\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star\|_F^2\le\frac{26\lambda_n^2R_q\tau^{-q}}{4}+4\lambda_nR_q\tau^{1-q}.
$$

To minimize the bound above, the best τ to pick is $\tau=26q(1-q)^{-1}\lambda_n/16$, which leads to

$$
\|\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star\|_F^2\le CR_q\lambda_n^{2-q}.
$$

Case 2. Suppose Case 1 fails so that we have

$$
\frac{1}{4n}\|\mathfrak{S}(\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star)\|_2^2\le\|\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star\|_F^2.
$$

(20) gives

$$
\frac{\lambda_n}{2}\|\Pi^\perp(\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star)\|_*\le2\lambda_n\|\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star\|_F+\frac{3\lambda_n}{2}\|\Pi(\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star)\|_*+2\lambda_n\|\Pi^\perp(\mathbf{\Gamma}^\star)\|_*,
$$

and from this, we can derive

$$
\|\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star\|_*\le8\sqrt{2r}\|\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star\|_F+4\|\Pi^\perp(\mathbf{\Gamma}^\star)\|_*.
$$

Case 2.1. Suppose

$$
\|\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star\|_F^2\le c_0(\sqrt{d_1d_2}\|\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star\|_\infty)\|\widehat{\mathbf{\Gamma}}-\mathbf{\Gamma}^\star\|_*\sqrt{\frac{d\log d}{n}}.
$$

Note that $\lambda_n\ge C\sqrt{d\log d/n}$ by definition. The equality might not directly hold due to the fitting error for $m(x)$. It is of great interest to see whether such equality can be attained with methods such as the cross-fitting technique to avoid double-dipping and reduce bias (Chernozhukov et al., 2018; Lu et al., 2025).

Now Using that

$$\sqrt{d_1 d_2}\|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}^\star\|_\infty \le 2a, \quad \|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}^\star\|_* \le 8\sqrt{2r}\|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}^\star\|_F + 4\|\Pi^\perp(\mathbf{\Gamma}^\star)\|_*,$$

and the choice of $\tau = C_q\lambda_n$ in Case 1, we obtain the inequality

$$\|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}^\star\|_F^2 \le CR_q^{1/2}\lambda_n^{1-q/2}\|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}^\star\|_F + CR_q\lambda_n^{2-q}.$$

Applying Young's inequality again, we obtain

$$\|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}^\star\|_F^2 \le CR_q\lambda_n^{2-q}.$$

Case 2.2. Suppose we have that

$$\frac{384d}{\sqrt{n}} \cdot \frac{\|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}^\star\|_\infty}{\|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}^\star\|_F} > \frac{1}{2},$$

then $\|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}^\star\|_F \le 1536a/\sqrt{n}$ and we obtain the desired rate directly.

Case 2.3. Suppose we are under neither *Case 2.1* and *Case 2.2*. Then the following inequality holds with probability at least $1 - c_1' \exp(-c_2' d \log d)$ by our proof in Step 4:

$$\frac{1}{\sqrt{n}}\|\mathfrak{S}(\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}^\star)\|_2 \ge \frac{1}{8}\|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}^\star\|_F - \frac{48d\|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}^\star\|_\infty}{\sqrt{n}} \ge \frac{1}{16}\|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}^\star\|_F.$$

Now using (20), we obtain that

$$\begin{aligned}\frac{1}{512}\|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}^\star\|_F^2 \le& 2\lambda_n\|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}^\star\|_F + \frac{3\lambda_n}{2}\|\Pi(\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}^\star)\|_* + 2\lambda_n\|\Pi^\perp(\mathbf{\Gamma}^\star)\|_* \\ \le& 3\sqrt{2r}\lambda_n\|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}^\star\|_F + 2\lambda_n\|\Pi^\perp(\mathbf{\Gamma}^\star)\|_*.\end{aligned}$$

Combining the choice of τ in Case 1, (18) and (19), with Young's inequality, we have

$$\|\widehat{\mathbf{\Gamma}} - \mathbf{\Gamma}^\star\|_F^2 \le CR_q\lambda_n^{2-q}.$$

Step 4. Justify the RSC condition. It remains now to justify the RSC condition. We prove the following result holds with high probability:

$$\frac{1}{\sqrt{n}}\|\mathfrak{S}(\mathbf{\Delta})\|_2 \ge \frac{1}{8}\|\mathbf{\Delta}\|_F - \frac{48d\|\mathbf{\Delta}\|_\infty}{\sqrt{n}}$$

for all $\mathbf{\Delta} \in \mathbb{R}^{d_1 \times d_2}$ in the set

$$\mathcal{C} = \left\{\mathbf{\Delta} \in \mathbb{R}^{d_1\times d_2} \mid \|\mathbf{\Delta}\|_F^2 \ge c_0(\sqrt{d_1 d_2}\|\mathbf{\Delta}\|_\infty)\|\mathbf{\Delta}\|_*\sqrt{\frac{d\log d}{n}}\right\}.$$

Define the following bad event:

$$\mathcal{B} = \left\{\exists \mathbf{\Delta} \in \mathcal{C} \mid \left|\frac{1}{\sqrt{n}}\|\mathfrak{S}(\mathbf{\Delta})\|_2 - \|\mathbf{\Delta}\|_F\right| > \frac{7}{8}\|\mathbf{\Delta}\|_F + \frac{48Ld\|\mathbf{\Delta}\|_\infty}{\sqrt{n}}\right\}.$$

Now we peel the set $\mathcal{C}$ with different radius:

$$\mathcal{C}(D) = \{\mathbf{\Delta} \in \mathcal{C} \mid \|\mathbf{\Delta}\|_\infty = d^{-1}, \|\mathbf{\Delta}\|_F \le D, \|\mathbf{\Delta}\|_* \le D^2/(c_0\sqrt{d\log d/n})\}$$

and the peeled event

$$\mathcal{B}(D) = \left\{\exists \mathbf{\Delta} \in \mathcal{C}(D) \mid \left|\frac{1}{\sqrt{n}}\|\mathfrak{S}(\mathbf{\Gamma})\|_2 - \|\mathbf{\Delta}\|_F\right| \ge \frac{3}{4}D + \frac{48L}{\sqrt{n}}\right\}.$$

By a discretization argument, Lemma 4 and Lemma 5 in Negahban and Wainwright (2012) proved the following bound: with probability at least $1 - 4\exp(-cnD^2)$, it holds that

$$\sup_{\mathbf{\Delta}\in\mathcal{B}(D)}\left|\frac{1}{\sqrt{n}}\|\mathfrak{S}(\mathbf{\Delta})\|_2 - \|\mathbf{\Delta}\|_F\right| \le \frac{3}{4}D + \frac{48L}{\sqrt{n}}.$$

Now observing that for any $\mathbf{\Delta} \in \mathcal{C}$ with $\|\mathbf{\Delta}\|_\infty = d^{-1}$, we have that

$$\|\mathbf{\Delta}\|_F^2 \geq c_0(\sqrt{d_1 d_2}\|\mathbf{\Delta}\|_\infty)\|\mathbf{\Delta}\|_* \sqrt{\frac{d \log d}{n}} \geq c_0 \|\mathbf{\Delta}\|_* \sqrt{\frac{d \log d}{n}} \geq c_0 \|\mathbf{\Delta}\|_F \sqrt{\frac{d \log d}{n}}.$$

This then leads to

$$\|\mathbf{\Delta}\|_F \geq c_0 \sqrt{\frac{d \log d}{n}}.$$

Therefore, we only need to focus on $\mathcal{C}'$ where

$$\mathcal{C}' = \{\mathbf{\Delta} \in \mathcal{C} \mid \|\mathbf{\Delta}\| = d^{-1}, \|\mathbf{\Delta}\|_F \geq c_0 \sqrt{d \log d/n}\}.$$

By setting $\alpha = 7/6$ and $v = c_0\sqrt{d \log d/n}$, we can peel $\mathcal{C}'$ into the following layers:

$$\mathcal{C}'_l = \{\mathbf{\Delta} \in \mathcal{C} \mid \|\mathbf{\Delta}\|_\infty = d^{-1}, \alpha^{l-1} v \leq \|\mathbf{\Delta}\|_F \leq \alpha^l v, \|\mathbf{\Delta}\|_* \leq (\alpha^l v)^2/(c_0\sqrt{d \log d/n})\}.$$

If the bad event $\mathcal{B}$ holds for some matrix $\mathbf{\Delta}$, then $\mathbf{\Delta}$ must belong to some set $\mathcal{C}'_l$, which satisfies

$$\left| \frac{1}{\sqrt{n}} \|\mathfrak{S}(\mathbf{\Delta})\|_2 - \|\mathbf{\Delta}\|_F \right| > \frac{7}{8}\alpha^{l-1} v + \frac{48L}{\sqrt{n}} = \frac{3}{4}\alpha^l v + \frac{48L}{\sqrt{n}}.$$

Thus, $\mathcal{B}(\alpha^l v)$ must hold. Applying a union bound, we must have

$$\begin{aligned}
\mathbb{P}\{\mathcal{B}\} \leq & \sum_{l=1}^{\infty} \mathbb{P}\left\{\mathcal{B}(\alpha^l v)\right\} \leq c_1 \sum_{l=1}^{\infty} \exp(-c_2 n \alpha^{2l} v^2) \\
\leq & c_1 \sum_{l=1}^{\infty} \exp(-2c_2 n \log(\alpha) l d \log d) \\
= & \frac{c_1 \exp(-c'_2 d \log d)}{1 - \exp(-c'_2 d \log d)} \\
\lesssim & c'_1 \exp(-c'_2 d \log d).
\end{aligned}$$

□

D.5 Proof of Theorem 5.1

Proof of Theorem 5.1. In the first stage, the regret is bounded by

$$\text{Regret}_e = \sum_{t=1}^{T_e} e_{z^\star,t}^\top \mathbf{\Gamma}^\star e_{x,t} - e_{\hat{z},t}^\top \mathbf{\Gamma}^\star e_{x,t} \leq 2T_e \cdot \|\mathbf{\Gamma}^\star\|_\infty.$$

In the second stage, the regret is

$$\begin{aligned}
\text{Regret}_c = & \sum_{t=T_e+1}^{T} e_{z,t}^{\star\top} \mathbf{\Gamma}^\star e_{x,t}^\star - \widehat{e}_{z,t}^\top \mathbf{\Gamma}^\star e_{x,t}^\star \\
= & \sum_{t=T_e+1}^{T} (e_{z,t}^\star - \widehat{e}_{z,t})^\top (\mathbf{\Gamma}^\star - \widehat{\mathbf{\Gamma}}) e_{x,t}^\star + (e_{z,t}^\star - \widehat{e}_{z,t})^\top \widehat{\mathbf{\Gamma}} e_{x,t}^\star \\
\leq & \sum_{t=T_e+1}^{T} (e_{z,t}^\star - \widehat{e}_{z,t})^\top (\mathbf{\Gamma}^\star - \widehat{\mathbf{\Gamma}}) e_{x,t}^\star \\
\leq & 2T_c \cdot \|\mathbf{\Gamma}^\star - \widehat{\mathbf{\Gamma}}\|_\infty \\
\leq & 2T_c \cdot \|\mathbf{\Gamma}^\star - \widehat{\mathbf{\Gamma}}\|_F \\
\leq & 2T_c \left(\frac{d_1 d_2 \log(d_1 + d_2) r d}{T_e} \right)^{1/2}.
\end{aligned}$$

Therefore, the total regret across the two stages is

$$\begin{aligned}\text{Regret} =& \text{Regret}_e + \text{Regret}_c \\ \leq& 2T_e \cdot \|\mathbf{\Gamma}^\star\|_\infty + 2(T - T_e) \cdot \|\mathbf{\Gamma}^\star\|_\infty \cdot \left(\frac{d_1 d_2 \log(d_1 + d_2) r d}{T_e}\right)^{1/2}.\end{aligned}$$

Now setting $T_e = T^a$ for some $a \in (0, 1)$. The above upper bound is minimized at

$$a \asymp \frac{1}{3} \log_T \{CT^2 d_1 d_2 \log(d_1 + d_2) r d\},$$

which leads to a final regret bound

$$\text{Regret} \leq CT^{2/3} d \log d^{1/3} r^{1/3}.$$

□

E Additional simulation results

E.1 Comparison of several kernel-based estimation strategies

Based on the setup of the Upworthy experiment, we also conduct a Monte Carlo simulation to evaluate the training and testing error of the proposed method. For the first experiment, we evaluate the training and testing error of the proposed method and compare it with several baseline methods. To do this, we split the synthetic dataset into a training set and a testing set, fit the training data with different methods, and evaluate the fitted results on the testing dataset. The four baseline methods we evaluate are: (i) double kernel representation learning (`DKRL`); (ii) RKHS regression with only treatments (`Treatment Only`); (iii) RKHS regression with only covariates (`Covariate Only`); (iv) RKHS regression with both treatments and covariates using a product kernel $\mathcal{K}_g \odot \mathcal{K}_h$ (`Product Kernel`). In our simulation, we vary across multiple levels of regularization parameters and compute the mean squared training and testing error. The results are summarized in Figure 2.

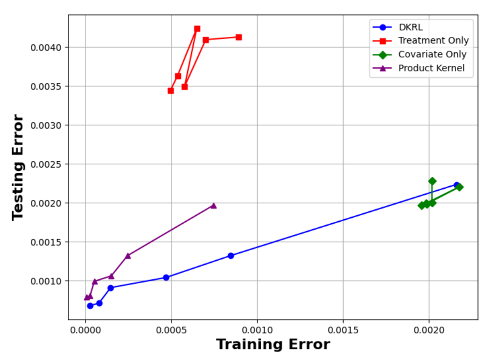


Figure 2: Training and testing error for different methods

E.2 Interpretability of DKRL results

To further elaborate on the interpretability of the learned feature, we show several example headlines with high or low semantic scores in Table 2, which are good demonstrations on how the features capture the semantic meanings.

On the X row representing "sharp contrasts with narrative surprises", high-scored examples carry stronger “contrast + surprise” hooks. Each headline pairs a familiar setup with a jarring twist—Hank Green sparring with a “corporate suit” over a supposedly dull yet vital topic, a Black cop repeatedly stop-and-frisked, and cute animals foreshadowing a darker reveal—creating instant narrative surprise. The low-scored examples, on the contrary, lean more on curiosity or emotional uplift (e.g., “A Lot Of

You Will Want To Know About It Anyway", "hese Pics Of A Baby Bird's Rescue Are So Moving") and less on jarring juxtapositions. They intrigue, but they don't rely as heavily on opposites twists.

On the Y row representing "conflict-driven tone", high-scored examples signal confrontation, outrage, or direct criticism, with words like "Deport THIS GUY", "attacks", "smackdown", "bad-ass", "tears into the awful folks". Besides, the examples pit one party against another (Hank Green vs. Corporate Suit, pink gang vs. predators, Fox Anchor vs. colleagues). The low-scored examples lean toward curiosity, wonder, and mild social critique. It uses softer hooks ("made me the happiest", "reason to think twice") and lacks the adversarial or combative language that characterizes the high-scored ones.

Table 2: Interpretability of the learned representations from DKRL in the Upworthy experiment. Feature X (X-axis in Figure 1(a)) represents whether there are "sharp contrasts with narrative surprises", and feature Y (Y-axis in Figure 1(a)) represents whether the sentence is in a "conflict-driven tone".

Feature	Example headlines with high scores	Example headlines with low scores
X	- "Hank Green Goes Up Against A Corporate Suit To Debate An Unsexy Issue That Is Really A Big Deal"; - "He's Black. He's a Cop. He's Also Been Stopped and Frisked 30 Times."; - "Floppy Ears, Soulful Eyes, Adorable Babies. And They Have Something Else People Can't Stop Taking."	- "Anyone Able To Read This Won't Benefit From It, But A Lot Of You Will Want To Know About It Anyway"; - "She Won A Golden Globe And Dedicated It To A Teen She Never Met. Here's Why It's A Big Deal."; - "These Pics Of A Baby Bird's Rescue Are So Moving You Might Get Emotionally Involved Here"
Y	- "Hey, I Have An Idea: Let's Deport THIS GUY, Instead"; - "This Pink Gang Attacks Sexual Predators And Rapists That Harm Them. Tell Me, What Else Can They Do?"; - "PLOT TWIST: Fox News Anchor Lays Epic Smackdown On Fox News Anchors For Obvious And Blatant Misogyny"	- "A Tiny Probe Millions Of Miles Away Just Made Me The Happiest I've Been In Weeks"; - "Here's A List Of Uncool Problems For Ladies That We Could Fix To Make Life Easier For Everyone"; - "A Reason To Think Twice Before Buying Your Kid All Those LEGOs"

E.3 Sensitivity to hyperparameter tuning

We evaluated the sensitivity of our Algorithm 1 to two tuning parameters, penalty level λ and rank r, in Algorithm 1. The results are reported in Figure 3. In terms of selection of penalty level, an appropriate choice of penalty is important to balance the level of regularization and loss minimization. In practice, we recommend using cross validation to tune this parameter. In terms of rank selection, we can see that selecting a rank that is smaller than the truth could lead to bias, as this loses important features. In contrast, there is minimal bias from selecting a slightly larger rank. In practice, we recommend starting with a larger rank and trimming slightly.

E.4 The MIND dataset experiment

In this subsection, we report the results for the synthetic simulation we conducted for the MIND dataset in Table 3. As we have commented in the main paper, while most methods benefit from the low-rank structure, DKRL exploits the low-rank structure more effectively. And even with a high-rank interaction matrix, DKRL utilizes the similarity structure in the embeddings to enhance accuracy.

E.5 The ASOS experiment

In the second simulation based on the ASOS experiment, we generated a synthetic study following the similar DGP to the one used in the main paper, with newly generated embeddings from the

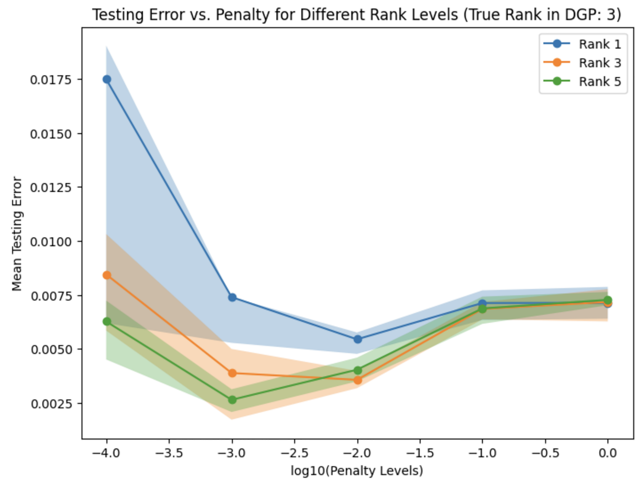


Figure 3: Hyperparameter sensitivity against penalty level λ and rank r.

Table 3: Testing MSE across methods (rows) and decay levels (columns)

Method	$q = 0$	$q = 2$	$q = 4$	$q = 6$
DKRL	**0.007 (0.003)**	**0.003 (0.001)**	**0.003 (0.001)**	**0.003 (0.001)**
Lasso_Z	0.034 (0.003)	0.009 (0.002)	0.008 (0.002)	0.008 (0.002)
XGB_Z	0.036 (0.003)	0.007 (0.002)	0.007 (0.002)	0.007 (0.002)
FNN_Z	0.034 (0.003)	0.011 (0.002)	0.011 (0.002)	0.011 (0.002)
Kernel_Z	0.031 (0.003)	0.006 (0.002)	0.006 (0.001)	0.006 (0.002)
Lasso_X	0.055 (0.005)	0.013 (0.004)	0.011 (0.003)	0.011 (0.003)
XGB_X	0.056 (0.006)	0.014 (0.004)	0.011 (0.003)	0.011 (0.003)
FNN_X	0.055 (0.005)	0.014 (0.004)	0.011 (0.003)	0.011 (0.003)
Kernel_X	0.055 (0.005)	0.013 (0.004)	0.011 (0.003)	0.011 (0.003)
Lasso_ZX	0.032 (0.003)	0.009 (0.001)	0.008 (0.002)	0.008 (0.002)
XGB_ZX	0.022 (0.001)	0.004 (0.001)	0.004 (0.001)	0.004 (0.001)
FNN_ZX	0.016 (0.002)	0.010 (0.002)	0.009 (0.002)	0.010 (0.002)
Kernel_ZX	0.035 (0.002)	0.008 (0.002)	0.007 (0.001)	0.007 (0.002)

shopping item information from the ASOS dataset. We then compared the KDRL method with two other methods: deep-learning based (SIN) method and product-kernel based (ProdKernel) method in terms of their training/testing error and computation time. The results are reported in Table 4, from which we consolidate the finding that DKRL performs fairly satisfactorily in terms of both training/testing errors and computation time.

Table 4: ASOS Experiment by Rank and Method (Mean (Std), rounded to four decimals)

Rank	Method	Train Error	Test Error	Time
2	`SIN`	0.0094 (0.0004)	0.0095 (0.0011)	7.9361 (0.4858)
	`DKRL`	**0.0022 (0.0002)**	**0.0035 (0.0009)**	0.1739 (0.0305)
	`ProdKernel`	0.0036 (0.0001)	0.0062 (0.0010)	0.0013 (0.0002)
3	`SIN`	0.0085 (0.0004)	0.0085 (0.0011)	8.0929 (0.3153)
	`DKRL`	**0.0025 (0.0002)**	**0.0047 (0.0010)**	0.4702 (0.1094)
	`ProdKernel`	0.0044 (0.0002)	0.0075 (0.0012)	0.0014 (0.0003)
5	`SIN`	0.0119 (0.0014)	0.0120 (0.0014)	7.9983 (0.4951)
	`DKRL`	**0.0027 (0.0000)**	**0.0076 (0.0012)**	1.6868 (0.3907)
	`ProdKernel`	0.0057 (0.0000)	0.0100 (0.0014)	0.0014 (0.0004)
10	`SIN`	0.0175 (0.0007)	0.0175 (0.0018)	8.1151 (0.4709)
	`DKRL`	**0.0026 (0.0001)**	**0.0128 (0.0017)**	7.9518 (0.8747)
	`ProdKernel`	0.0083 (0.0003)	0.0147 (0.0017)	0.0017 (0.0004)

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: we outlined our contributions in the abstract and Section 1. Section 3-6 fully elaborates on all the points we made.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: we discussed the limitation of the work in Conclusion (Section 7).

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.

- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: we presented the full sets of assumptions in the statement of the Theorems (Theorem 4.1, 4.2, 5.1), and provide proofs to all the claims we made in Appendix D.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: we include the anonymized version of our code as supplementary materials and we are happy to release the code and data once accepted. Our main experimental results mainly involve an implementation of our algorithms, which is detailed in 1 and 2, and running them on synthetic datasets which are publicly available as detailed in Section 6 of the paper.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same

dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.

- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We submit an anonymized version of our code as supplementary materials.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: These are discussed in Section 6 of the main paper.

   Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: we include standard deviations or inter-quantile ranges for the experiment results.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
   - If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: we used personal computers (MBP with a M2 Max chip) to run the experiments on CPU and did not use additional computing resources. This information is included in Section 6.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

Answer: [Yes]

Justification: we follow the NeurIPS Code of Ethics.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: we outlined the social impacts of the paper in advancing the techniques for digital experimentation in our contribution. Yet there could negative impacts related to the use of Large Language Models in experimentation such as privacy and fairness issues. We also outlined this in the Conclusion section.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: the paper poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.

- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [NA]

    Justification: the paper does not use existing assets.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: the paper does not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: the paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.

- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: the paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: we discussed the use of LLMs as this is a core part of our methodology.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.