A Appendix

A.1 Bounds for $R(\mathcal{Z}|\mathbf{K})$

We prove the general bounds for $R(\mathcal{Z}|\mathbf{K})$ by proving the lower and upper bound independently using the following lemmas.

Lemma 3 (Lower bound for $R(\mathcal{Z}|\mathbf{K})$)**.** *For $\mathcal{Z} \in \mathbb{R}^{n\times d}$, RBF kernel matrix $\mathbf{K} \in \mathbb{R}^{n\times n}$ using a distance function $\mathsf{d}(\cdot,\cdot)$ that satisfies $\mathsf{d}(x,x) = 0$ and $\epsilon > 0$, it holds that:*

$$R(\mathcal{Z}|\mathbf{K}) \geq R(\mathcal{Z}) \tag{7}$$

where the equality is satisfied only when $\mathbf{K} = \mathbf{1}\mathbf{1}^T$.

Proof. We start off by writing down the expanded form of $R(\mathcal{Z}|\mathbf{K})$:

$$R(\mathcal{Z}|\mathbf{K}) = \frac{1}{2}\log_2\det\left(I + \frac{d}{n\epsilon^2}\mathcal{Z}\mathcal{Z}^T \odot \mathbf{K}\right) \tag{8}$$

In the above equation 8, we first note that both $\mathcal{Z}\mathcal{Z}^T$ and $\mathbf{K}$ matrices are positive-semi definite symmetric matrices. Using Schur product theorem [49], we can show that their hadamard product $\mathcal{Z}\mathcal{Z}^T \odot \mathbf{K}$ is also positive semi-definite (for $d > 1$). Next, we utilize the following property for Hadamard products:

Theorem 7.25 [48]. Given two positive semi-definite square matrices A and B of dimension m. Then, the following property holds: $\det(A \odot B) \geq \det(A)\prod_{i=1}^{m} b_{ii}$

Applying this property to $\mathcal{Z}\mathcal{Z}^T \odot \mathbf{K}$, we get the following:

$$\det(\mathcal{Z}\mathcal{Z}^T \odot \mathbf{K}) \geq \det(\mathcal{Z}\mathcal{Z}^T) \tag{9}$$

where $\mathbf{K}_{ii} = 1, \forall i$, as it is a RBF kernel and $\mathsf{d}(i,i) = 0$. Now, since $\mathcal{Z}\mathcal{Z}^T \odot \mathbf{K}$ and $\mathcal{Z}\mathcal{Z}^T$ are positive semi-definite, their corresponding eigenvalues are non-negative, $\lambda_i(\mathcal{Z}\mathcal{Z}^T \odot \mathbf{K}) \geq 0$ and $\lambda_i(\mathcal{Z}\mathcal{Z}^T) \geq 0$. Since the eigenvalues are non-negative, we can extend Equation 4 as follows:

$$\begin{aligned}
\prod_i \lambda_i(\mathcal{Z}\mathcal{Z}^T \odot \mathbf{K}) &\geq \prod_i \lambda_i(\mathcal{Z}\mathcal{Z}^T) \\
\prod_i \left(1 + \frac{d}{n\epsilon^2}\lambda_i(\mathcal{Z}\mathcal{Z}^T \odot \mathbf{K})\right) &\geq \prod_i \left(1 + \frac{d}{n\epsilon^2}\lambda_i(\mathcal{Z}\mathcal{Z}^T)\right) \\
\det\left(I + \frac{d}{n\epsilon^2}\mathcal{Z}\mathcal{Z}^T \odot \mathbf{K}\right) &\geq \det\left(I + \frac{d}{n\epsilon^2}\mathcal{Z}\mathcal{Z}^T\right) \\
R(\mathcal{Z}|\mathbf{K}) &\geq R(\mathcal{Z})
\end{aligned} \tag{10}$$

where the second inequality holds because the affine transform of positive variables preserves inequalities. The equality is satisfied when $\mathbf{K} = \mathbf{1}\mathbf{1}^T$. □

Lemma 4 (Upper bound for $R(\mathcal{Z}|\mathbf{K})$)**.** *For $\mathcal{Z} \in \mathbb{R}^{n\times d}$, RBF kernel matrix $\mathbf{K} \in \mathbb{R}^{n\times n}$ using a distance function $\mathsf{d}(\cdot,\cdot)$ that satisfies $\mathsf{d}(x,y) = \mathsf{d}(y,x)$ and and $\epsilon > 0$, it holds that:*

$$R(\mathcal{Z}|\mathbf{K}) \leq \frac{n}{2}\log_2\left(1 + d/n\epsilon^2\right) \tag{11}$$

Proof. We start by noting that the Hadamard product of two positive semi-definite matrices $\mathcal{Z}\mathcal{Z}^T \odot \mathbf{K} \in \mathbb{R}^{n\times n}$ is positive semi-definite (using the Schur product theorem). We also assume that the representations $z_i \in \mathcal{Z}$ are unit normalized, thereby the diagonal entries of $(\mathcal{Z}\mathcal{Z}^T)_{ii} = 1$. The diagonal entries $\mathbf{K}_{ii} = 1$ as $\mathsf{d}(z_i, z_i) = 0$, which implies $(\mathcal{Z}\mathcal{Z}^T \odot \mathbf{K})_{ii} = 1$. Given these facts, we can write the following properties of about the eigenvalues of $\mathcal{Z}\mathcal{Z}^T \odot \mathbf{K}$:

$$\lambda_i(\mathcal{Z}\mathcal{Z}^T \odot \mathbf{K}) \geq 0, \sum_i \lambda_i(\mathcal{Z}\mathcal{Z}^T \odot \mathbf{K}) = n \tag{12}$$

where the second property follows from the fact that $\mathrm{tr}(\mathcal{Z}\mathcal{Z}^T \odot \mathbf{K}) = n$. We are interested in finding the maximum value of $R(\mathcal{Z}|\mathbf{K})$ that can be written as:

$$R(\mathcal{Z}|\mathbf{K}) = \frac{1}{2}\log_2 \prod_{i=1}^{n} \left(1 + \frac{d}{n\epsilon^2}\lambda_i(\mathcal{Z}\mathcal{Z}^T \odot \mathbf{K})\right) \tag{13}$$

To maximize $R(\mathcal{Z}|\mathbf{K})$, we need to maximize the product within the logarithm. Each term within the product $1 + \frac{d}{n\epsilon^2}\lambda_i(\mathcal{Z}\mathcal{Z}^T \odot \mathbf{K}) \geq 0$ (eigenvalues of a PSD matrix). Using the AM-GM inequality, the product is maximized when all the individual terms are equal.

$$\lambda_i(\mathcal{Z}\mathcal{Z}^T \odot \mathbf{K}) = n/n = 1 \tag{14}$$

Substituting this result in Equation 13, we obtain the following upper bound:

$$R(\mathcal{Z}|\mathbf{K}) \leq \frac{n}{2}\log_2(1 + d/n\epsilon^2) \tag{15}$$

where the equality is achieved when $\mathcal{Z}\mathcal{Z}^T = I$ when all the representations are orthogonal. Note that this is only possible when $d \geq n$.

□

A.2 Proof of Lemma 2

Lemma 2 (Alignment for random representations)**.** *Expected $A_k(f)$ score achieved by a concept erasure framework f that generates random representations is $\mathbb{E}[A_k(f)] = k/n$.*

Proof. To prove this, we first assume two randomly generated k-nearest neighbour graphs (since the original representation is uncorrelated with the randomly generated one we can consider it's random). As it is a kNN graph, for each node has an expected degree $\mathbb{E}[d] \approx k$, where d is the degree of the node. Now, let's consider the probability of a node x_i being part of node x_j:

$$\begin{aligned} p(x_i \in \mathrm{knn}(x_j)) &= \frac{d_i}{n} \\ \mathbb{E}[p(x_i \in \mathrm{knn}(x_j))] &= \frac{\mathbb{E}[d_i]}{n} = \frac{k}{n} \end{aligned} \tag{16}$$

where d_i is the degree of node i and n is the total number of representations. Since computing the exact probability requires knowledge of the degree of the node, we compute the expectation of the same. Next, we compute the probability that node i is present in both kNN sets (before and after debiasing) of node j:

$$\begin{aligned} \mathbb{E}[\mathrm{knn}(x) \cap \mathrm{knn}(z)] &= \mathbb{E}\left[\sum_j p(x_i \in \mathrm{knn}(x_j) \wedge z_i \in \mathrm{knn}(z_j))\right] \\ &= \sum_j \mathbb{E}\left[p(x_i \in \mathrm{knn}(x_j))p(z_i \in \mathrm{knn}(z_j))\right] \\ &= \sum_j \mathbb{E}\left[p(x_i \in \mathrm{knn}(x_j))\right]\mathbb{E}\left[p(z_i \in \mathrm{knn}(z_j))\right] \\ &= \sum_{j=1}^{n} k^2/n^2 = k^2/n \end{aligned} \tag{17}$$

where the first step utilizes linearity of expectation, and the second step follows from the fact that the degree of distribution of $\mathcal{X}$ and $\mathcal{Z}$ are independent. Replacing the result from Eqn 17 in Eqn 6, we get $A_k(f) = k/n$. □

A.3 Proof of Lemma 3

Lemma 3 (Upper Bound for categorical concepts)**.** *For categorical concept variables with the kernel values $\mathbf{K}_{ij} \in \{0, 1\}$, $R(\mathcal{Z}|\mathbf{K})$ is bounded by the sum of rate-distortion functions of representation set from individual classes $\mathcal{Z}_j$*

$$R(\mathcal{Z}|\mathbf{K}) = \sum_{j=1}^{m} \frac{1}{2} \log_2 \det \left(I + \frac{d}{n\epsilon^2} \mathcal{Z}_j \mathcal{Z}_j^T \right) \leq \sum_{j=1}^{m} R(\mathcal{Z}_j) \tag{18}$$

where the equality holds only when $\mathcal{Z}_j \mathcal{Z}_j^T = 0, \forall j$ and m is the number of classes.

Proof. For categorical variables, the kernel function takes the following form:

$$k(i, j) = \begin{cases} 1, & \text{if } a_i = a_j \\ 0, & \text{if } a_i \neq a_j \end{cases} \tag{19}$$

If the kernel function $k(\cdot, \cdot)$ is of the above form. Using the corresponding kernel matrix $\mathbf{K}$ we get,

$$\mathbf{M} = I + \frac{d}{n\epsilon^2} \mathcal{Z}\mathcal{Z}^T \odot \mathbf{K} = I + \frac{d}{n\epsilon^2} \begin{bmatrix} \mathcal{Z}_1 \mathcal{Z}_1^T & 0 & \dots & 0 \\ 0 & \mathcal{Z}_2 \mathcal{Z}_2^T & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \mathcal{Z}_k \mathcal{Z}_k^T \end{bmatrix} \tag{20}$$

where $\mathbf{M}$ becomes a block diagonal matrix and Z_i's are representations belonging to class i. Using the determinant property of block diagonal matrices, we have:

$$\begin{aligned} \log_2 \det(\mathbf{M}) &= \sum_{j=1}^{m} \log_2 \det \left(I + \frac{d}{n\epsilon^2} \mathcal{Z}_j \mathcal{Z}_j^T \right) \\ R(Z|\mathbf{K}) &= \sum_{j=1}^{m} \frac{1}{2} \log_2 \det \left(I + \frac{d}{n\epsilon^2} \mathcal{Z}_j \mathcal{Z}_j^T \right) \end{aligned} \tag{21}$$

The individual terms in the above summation are closely related to the rate-distortion function of representation belonging to each class, j, as shown below:

$$R(Z_j) = \frac{1}{2} \log_2 \det \left(I + \frac{d}{n_j \epsilon^2} Z_j Z_j^T \right) \tag{22}$$

where n_j is the number of representations in class j. Note, $n_j < n$, where n is the total number of representations. Using the property that multiplying a matrix with a scalar is equivalent to multiplying its eigenvalues with the same scale, and that $\mathcal{Z}_j \mathcal{Z}_j^T$ is a PSD matrix. We can show:

$$\begin{aligned} R(Z|\mathbf{K}) &\leq \sum_{j=1}^{m} \frac{1}{2} \log_2 \det \left(I + \frac{d}{n\epsilon^2} Z_j Z_j^T \right) \\ R(Z|\mathbf{K}) &\leq \sum_{j=1}^{m} R(\mathcal{Z}_j) \end{aligned} \tag{23}$$

□

Notice that this is closely related to the MCR^2 objective, which tries to learn discriminative subspaces for individual classes. For concept erasure, we aim for the opposite effect by making instances from the same class dissimilar by maximizing their rate-distortion function.

Algorithm 1 Correlation Computation Routine

Input: Input representation set $\mathcal{X} \in \mathbb{R}^{n \times d}$
$\mathcal{Y} = \text{sgn}(\mathcal{X} W_1 W_2)$ ▷ generate labels using random weights $W_2 \in \mathbb{R}^{d \times m}, W_1 \in \mathbb{R}^{m \times 1}$
$\mathbf{U}, \Sigma, \mathbf{V} = \text{svd}(\mathcal{X})$
$\mathcal{Z}_0 = \mathcal{X}$ ▷ Initializing the representations
$A = \{\}, \text{scores} = \{\}$ ▷ accuracy and alignment sets
for $i \in \{1, \ldots, d\}$ **do**
  $\mathbf{u} = \frac{\mathbf{V}^T(i)}{\|\mathbf{V}^T(i)\|}$ ▷ access the i-th column of $\mathbf{V}$
  $\mathbf{P}_i = \mathbf{I}_d - \mathbf{u}\mathbf{u}^T$ ▷ null space projection matrix
  $\mathcal{Z}_i = \mathcal{Z}_{i-1} \mathbf{P}_i$
  $A = A \cup \text{acc}(\mathcal{Z}_i, \mathcal{Y})$ ▷ compute accuracy
  $\text{scores} = \text{scores} \cup A_k(\prod \mathbf{P}_i)$ ▷ compute alignment scores
end for
$r = \text{Pearson}(A, \text{scores})$ ▷ compute the Pearson correlation
return r

B Alignment Scoring

In this section, we present several measures to capture information alignment and compare them with our proposed metric (in Section 4).

KSG MI estimator [33]. The Kraskov–Stogbauer–Grassberger (KSG) estimator uses the nearest neighbour information in the joint and marginal space to obtain a mutual information estimate. Specifically, it computes the number of neighbours around a point within a hypercube in the marginal spaces. The length of the hypercube is set based on the max-norm distance to the k-th neighbour in the joint space. The KSG MI estimate between two sets $\mathcal{X}$ and $\mathcal{Z}$ can be shown as follows:

$$I_{\text{KSG}}(\mathcal{X}, \mathcal{Z}) = \psi(k) - 1/k - \mathbb{E}[\psi(n_x) + \psi(n_z)] + \psi(N) \tag{24}$$

where $\psi(\cdot)$ is the digamma function, n_x and n_z are the number of points in the hypercube of the respective marginal spaces. In our experiment, we use the KSG MI estimator to evaluate the alignment between representation sets before and after concept erasure.

Degree distribution. In a k-nearest neighbour graph, some nodes are more connected to others (hub nodes) while others are sparsely connected. Building on our intuition of alignment A_k using the nearest neighbour graphs of representations, we can consider changes in its degree distribution, $D(\mathcal{X})$, during concept erasure to gauge how the underlying structure of the representation set has changed. We quantify the change using either L1-norm, L2-norm, or KL-divergence between the normalized degree distributions $D(\mathcal{X})$ and $D(\mathcal{Z})$.

Experiments. We perform experiments in a controlled setup to evaluate the efficacy of the proposed alignment measures.

(a) *Simulated Erasure*. In this experiment, we simulate knowledge erasure from a set of synthetic representations and observe how the alignment scores correlate with the downstream accuracy. Algorithm 1 shows the details for this process. First, we sample a set of representations from a uniform distribution $\mathcal{X} \sim \mathbb{R}^{n \times d}$ from a uniform distribution and construct a label set $\mathcal{Y}$ (using randomly sampled weights W_1, W_2). In a way, the label set retains some information about the original representations that we will probe as erasure happens. Then, we gradually remove information from representations $\mathcal{Z}$ by projecting them onto the nullspace $\mathbf{P}$ formed using the eigenvectors $\mathbf{u}$. After each iteration of projection, we compute the accuracy of predicting $\mathcal{Y}$ and alignment score, A_k. We report the Pearson correlation between the accuracies and information alignment in Table 3 (left side), along with the hyperparameter k used for each measure. We observe that A_k outperforms other approaches achieving better correlation, which showcases the efficacy of our approach.

(a) *Correlated Gaussians*. In this experiment, we sample two sets of Gaussians (zero mean) with a fixed covariance σ. In this setup, the mutual information has a closed-form solution:

Metric	Simulated Erasure k/n (%)	Simulated Erasure Pearson (r) ↑	Correlated Gaussian k/n (%)	Correlated Gaussian Pearson (r) ↑
KSG	10	0.965	0.02	**0.989**
KL-div (degree)	0.1	0.874	0.2	0.490
L2-norm (degree)	0.1	0.865	0.2	0.458
L1-norm (degree)	0.1	0.905	0.2	0.564
Alignment: A_k	50	**0.994**	50	0.969

Table 3: Comparison of A_k with other alignment measures on synthetic datasets. We observe that A_k achieves the best Pearson correlation scores with downstream accuracy on simulated concept erasure experiments due to the presence of a mapping function f. In a separate experiment, the KSG estimator shows the highest correlation with MI. A_k also achieves a high correlation score, while the degree distribution-based measures perform poorly due to the lack of a mapping function.

$$I(\mathcal{X}, \mathcal{Z}) = -\frac{1}{2}\log(1-\sigma^2) \tag{25}$$

We use the samples to compute the different alignment measures and investigate if they're correlated with the actual mutual information (Equation 25). Note that there does not exist an explicit mapping between these samples. In Table 3 (right side), we report the Pearson correlation scores for different measures. We find that the KSG MI estimator outperforms others, with A_k coming in as a close second. This is because our alignment scores assume a 1-to-1 mapping between the sets, which is absent in this case. The degree-distribution-based scores suffer even more as their measure is even more strongly reliant on the mapping. These results show that the alignment score A_k leverages the bijective mapping to generate scores that are well correlated with the mutual information but can be inaccurate in cases where the mapping function is absent.

C Implementation details

In this section, we provide various implementation details about our experimental setup. Specifically, we describe the details of the dataset, metrics, and hyperaparameters utilized.

C.1 Dataset

In this section, we describe the details of the datasets that were used in the experimental section.

GloVe embeddings [42]. We revisit the problem of deleting gender information (*binary attribute*) from word embeddings [12]. Specifically, we consider the GloVe embeddings of the 150k most frequently occurring words. For training `KRaM`, we follow the setup of [45, 17] to select the 7500 most male-biased, female-biased, and neutral words determined by the magnitude of the word vector's projection onto the gender direction (the largest principal component of the space of vectors formed using the difference gendered word vector pairs).

DIAL [11] is a Twitter-based sentiment classification dataset, where each tweet is associated with sentiment labels and "race" information (binary concept label) of the author. The sentiment concept labels are "happy" or "sad" and the binary race concept labels are "African-American English" (AAE) or "Standard American English" (SAE).

Synthetic dataset. We create a dataset where the representations are generated using a continuous latent variable, a, which serves as our concept label. During data generation, we first sample the latent variable $a \sim \text{Uni}(0, 1)$, and then sample the high-dimensional representation $x \sim \mathcal{N}(a\mathbf{1}_d, aI_d)$, where $\mathbf{1}_d$ is a vector of ones and I_d is the identity matrix. For this dataset, we set the dimension of the representations to be $d = 100$. In this setup, we observe that the latent concept label, a, is being used to scale the underlying isotropic Gaussian distribution. Therefore, post-concept erasure the representation space should appear like an isotropic Gaussian distribution, which is indeed the case as shown in Figure 5.

UCI Crimes [34]. This dataset[1] contains information about US communities in 1990 from various surveys. The dataset provides 128 attributes (both categorical and continuous variables) from 1,994 different US communities. we concatenate individual attributes of a community to obtain its representation. The regression task involves predicting the number of violent crimes per capita. We consider the ratio of African-American (AAE) people (*continuous* attribute) in a community as the concept to be erased.

Jigsaw Toxicity Classification [1]. This dataset contains online comments and the binary classification task involves detecting whether a comment is toxic or not. In this dataset, we consider two different concepts: *religion* and *race*. We consider a vector-valued protected attribute for this dataset. For the religion concept, we consider an unnormalized vector over the following categories: {'buddhist', 'christian', 'hindu', 'jewish', 'muslim', 'other_religion'}. Similarly, for the gender we consider the following categories: {'bisexual', 'female', 'heterosexual', 'homosexual, gay, or lesbian', 'male', 'other gender', 'other sexual orientation', 'transgender'}. During concept erasure of either concept, we only consider instances where at least one of the concept categories has a non-zero value and reserved 20% of the instances as the test set. This resulted in a dataset with a train/test split of (72k, 18k) for the religion concept and (106k, 26k) for the gender concept. We retrieve text representations for the comments from GPT-3.5 [14] and perform concept erasure on them.

C.2 Metrics

In this section, we present the details of the fairness metrics used in our experiments.

Demographic Parity (DP). Demographic Parity measures the difference in the probability of a prediction w.r.t to the protected attribute $\mathcal{A}$. Formally, it is defined as:

$$\text{DP} = \sum_{y \in \mathcal{Y}} |p(\hat{y} = y | \mathcal{A} = a) - p(\hat{y} = y | \mathcal{A} = \bar{a})| \tag{26}$$

where $a, \bar{a}$ are possible values of the binary concept and $\mathcal{Y}$ is the set of possible target attribute labels.

Generalized Demographic Parity (ΔGDP). Most of the literature on fairness metrics has focused on categorical variables. We use Generalized Demographic Parity (GDP) [31], which measures the discrepancy in outcome with respect to a continuous variable. GDP measure extends Demographic Parity for continuous protected attributes. It is defined as follows:

$$\Delta\text{GDP} = \int_0^1 |m(a) - m_{\text{avg}}| P(\mathcal{A} = a) da \tag{27}$$

where $m(a) = \mathbb{E}[\hat{y} | \mathcal{A} = a]$ is expected prediction of the model when protected attribute $\mathcal{A} = a$, $m_{\text{avg}} = \mathbb{E}[\hat{y}]$ is overall expected prediction, and $P(\mathcal{A} = a)$ is the probability that the protected attribute takes value a. The probability density $P(\cdot)$ can be measured using a histogram or kernel method. We used a kernel function to evaluate the probability density.

C.3 Hyperparameters

In our experiments, we primarily deal with two hyperparameters: regularization constant, λ (in Equation 4), and σ, associated with the standard deviation of a Gaussian kernel ($k(x, y) = e^{-\|x-y\|/\sigma^2}$). We set these parameters by performing a grid search on the development set using Weights & Biases [10]. We use a multi-layer neural network with ReLU non-linearity as the erasure function f. We further perform ablation experiments to understand the impact of these parameters on concept erasure performance (shown in Figure 8). All networks were trained using a single 22GB NVIDIA Quadro RTX 6000 GPU and experiments were executed in PyTorch [40] framework.

D Additional Results

In this section, we present additional experiments to analyze `KRaM`'s concept erasure performance.

[1] https://archive.ics.uci.edu/ml/datasets/Communities+and+Crime

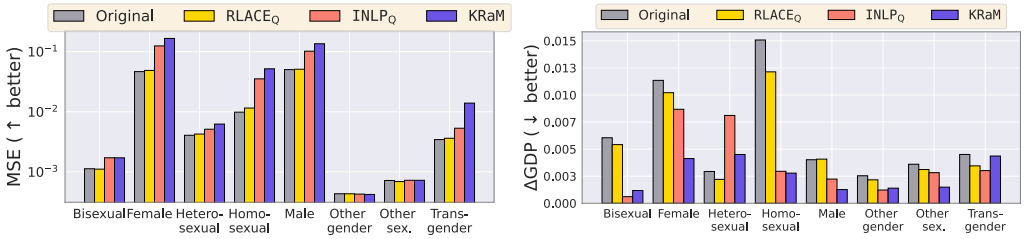

Figure 7: Vector-valued concept erasure performance using KRaM on Jigsaw toxicity classification dataset (gender concept). We observe a significant reduction in ΔGDP scores post erasure of vector-valued gender concept with negligible impact on toxicity classification performance.

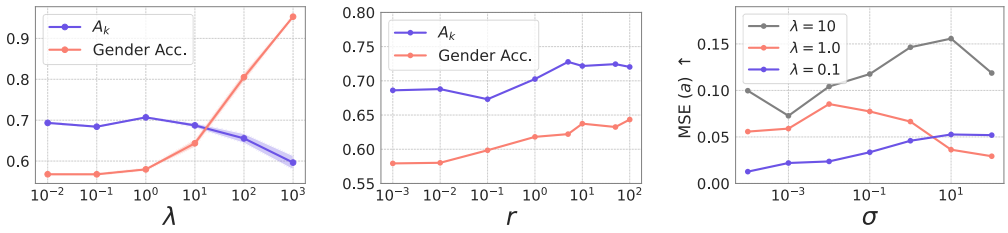

Figure 8: Ablation experiments to study the effect of parameters λ (Eqn. 4), r (a scaling factor in $R(\mathcal{Z}) = rb$), and σ (parameter in gaussian kernels) on the performance of concept deletion.

Vector-valued Concept Erasure. In this section, we present the results of vector-valued gender concept removal from GPT-3.5 text embeddings from the Jigsaw Toxicity classification dataset. We report the MSE and ΔGDP results in Figure 7. We observe that KRaM is able to significantly increase the gender MSE while simultaneously reducing the ΔGDP scores. During the debiasing process, we observe that there is minimal impact on the toxicity classification accuracy (91.9% → 90.1%).

Ablation with different kernels. We perform ablation experiments with different kernel functions used to define the $\mathbf{K}$ and observe its impact on the concept erasure performance. In Table 4, we report the results for erasing the continuous concept on the synthetic dataset. Apart from the kernel function, we use the same hyperparameters in all setups. We observe that KRaM achieves similar concept erasure performance using different kernel functions. We observe that using the Gaussian kernel function in KRaM yields the best erasure performance and alignment score A_k improves when we use a linear erasure function f.

Method	MSE (a) ↓	A_k ↑
Original	0.006	1.0
KRaM (Laplace)	0.083	0.68
KRaM (Cauchy)	0.092	0.63
KRaM$_{\text{linear}}$ (Gaussian)	0.083	**0.75**
KRaM (Gaussian)	**0.109**	0.67

Table 4: Ablations with kernel functions: we observe that KRaM achieves similar performance using different kernel functions.

Ablation of parameters. In this experiment, we perform ablations with several parameters in KRaM and observe how that affects the concept erasure performance. First, we experiment with gender removal from GloVe embeddings to understand the impact of λ (Eqn. 4). In Figure 8 (left), we observe that as λ increases, concept erasure worsens (↑ gender accuracy). This is expected as the erasure function f is penalized for $|R(\mathcal{Z}) - b|$ term more than maximizing $R(\mathcal{Z}|\mathbf{K})$ (which helps in erasure). The alignment scores A_k stay mostly stable with a minor drop at high λ values. We belive this happens as f aims to match the rate-distortion constant, possibly neglecting the underlying representation structure. Second, in the same setup, we modify the equality constraint to be: $|R(\mathcal{Z}) - rb|$ and ablate r (shown in Figure 8 (center)). We observe that both alignment scores and gender accuracy increase with an increase in r, which demonstrates the importance of this constraint. Even though $R(\mathcal{Z}|\mathbf{K})$ is maximized, if the overall feature space expands (high r), the concept variable can still become distinguishable (high gender accuracy). Third, in Figure 8 (right), we report the MSE scores on the synthetic dataset for varying σ (the parameter in the gaussian kernel). In all setups within Figure 8 (right), we notice the same pattern of increasing MSE (a) scores followed by a decrease. We believe

this drop happens with higher σ values because distances become very small and kernel values are similar. This results in the kernel ignoring the similarity of instances in the concept space.

E Broader Impact & Limitations

In this section, we discuss the broader societal impact and limitations of our framework, KRaM.

Limitations. While erasing sensitive concept attributes can reduce bias and improve privacy, it may also result in the loss of potentially useful information for the task at hand. This could negatively impact the utility of the model. The definition of what constitutes a sensitive concept attribute can vary greatly depending on the cultural, ethical, and legal context. This work assumes that these sensitive attributes can be clearly defined and agreed upon, which might not always be the case. Therefore, developers using such erasure frameworks should take care of the societal impact before utilizing them in the wild.

Negative Usage. KRaM s intended to be used in scenarios where the user is already aware of the concept attribute to be erased. KRaM can only be trained on data where concept labels are annotated either as categorical, continuous, or vector-valued attributes. One potential misuse of KRaM would be to define relevant features for a task (e.g., experience for a job application) as a concept to be erased. In such cases, the classification system may be forced to rely on sensitive demographic information for predictions. It is possible to flag systems in these cases by evaluating the statistical parity when the concept attributes have changed.

In general, we hope that our proposed concept erasure framework, KRaM, would encourage others to develop more robust concept erasure systems that can simultaneously retain a lot of information from the original representations.