Journal of Machine Learning Research 25 (2024) 1-76 Submitted 9/23; Revised 1/24; Published 5/24

Neural Feature Learning in Function Space*

Xiangxiang Xu XUXX@MIT.EDU

Lizhong Zheng LIZHONG@MIT.EDU

Department of Electrical Engineering and Computer Science
Massachusetts Institute of Technology
Cambridge, MA 02139-4307, USA

Editor: Kilian Weinberger

Abstract

We present a novel framework for learning system design with neural feature extractors. First, we introduce the feature geometry, which unifies statistical dependence and feature representations in a function space equipped with inner products. This connection defines function-space concepts on statistical dependence, such as norms, orthogonal projection, and spectral decomposition, exhibiting clear operational meanings. In particular, we associate each learning setting with a dependence component and formulate learning tasks as finding corresponding feature approximations. We propose a nesting technique, which provides systematic algorithm designs for learning the optimal features from data samples with off-the-shelf network architectures and optimizers. We further demonstrate multivariate learning applications, including conditional inference and multimodal learning, where we present the optimal features and reveal their connections to classical approaches.

Keywords: Feature geometry, information processing, neural feature learning, nesting technique, multivariate dependence decomposition

1. Introduction

Learning useful feature representations from data observations is a fundamental task in machine learning. Early developments of such algorithms focused on learning optimal linear features, e.g., linear regression, PCA (Principal Component Analysis) (Pearson, 1901), CCA (Canonical Correlation Analysis) (Hotelling, 1936), and LDA (Linear Discriminant Analysis) (Fisher, 1936). The resulting algorithms admit straightforward implementations, with well-established connections between learned features and statistical behaviors of data samples. However, practical learning applications often involve data with complex structures which linear features fail to capture. To address such problems, practitioners employ more complicated feature designs and build inference models based on these features, e.g., kernel methods (Cortes and Vapnik, 1995; Hofmann et al., 2008) and deep neural networks (LeCun et al., 2015). The feature representations serve as the information carrier, capturing useful information from data for subsequent processing. An illustration of such feature-centric learning systems is shown in Figure 1, which consists of two parts:

1. A *learning* module which generates a collection of features from the data. Data can take different forms, for example, input-output pairs[1] or some tuples. The features can be either specified implicitly, e.g., by a kernel function in kernel methods, or explicitly parameterized

*. This work was presented in part at 2022 58th Annual Allerton Conference on Communication, Control, and Computing (Allerton), Monticello, IL, USA, Sep. 2022 (Xu and Zheng, 2022).

1. In literature, the input variables are sometimes referred to as independent/predictor variables, and the output variables are also referred to as dependent/response/target variables.

as feature extractors, e.g., artificial neural networks. The features are learned via a training process, e.g., optimizing a training objective defined on the features.

2. An *assembling* module which uses learned features to build an inference model or a collection of inference models. The inference models are used to provide information about data. For example, when the data take the form of input-output pairs, a typical inference model provides prediction or estimation of output variables based on the input variables. The assembling module determines the relation between features and resulting models, which can also be specified implicitly, e.g., in kernel methods.

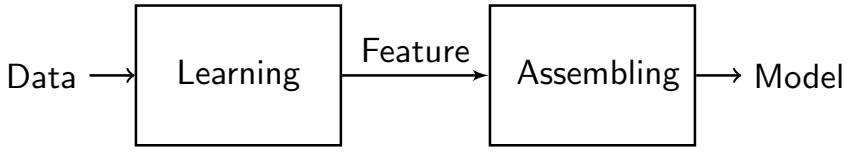


Figure 1: Schematic diagram of a general feature-centric learning system

Learning systems are commonly designed with a predetermined assembling module, which allows the whole system to be learned in an end-to-end manner. One representative example of such designs is deep neural networks. On one hand, this end-to-end characteristic makes it possible to employ large-scale and often over-parametrized neural feature extractors (LeCun et al., 2015; Krizhevsky et al., 2017; He et al., 2016; Vaswani et al., 2017), which can effectively capture hidden structures in data. On the other hand, the choices of assembling modules and learning objectives are often empirically designed, with design heuristics varying a lot across different tasks. Such heuristic choices make the learned feature extractors hard to interpret and adapt, often viewed as black boxes. In addition, the empirical designs are typically inefficient, especially for multivariate learning problems, e.g., multimodal learning (Ngiam et al., 2011), where there can exist many potentially useful assembling modules and learning objectives to consider.

To address this issue and obtain more principled designs, recent developments have adopted statistical and information-theoretical tools in designing training objectives. The design goal is to guarantee that learned features are informative, i.e., carry useful information for the inference tasks. To this end, a common practice is to incorporate information measures in learning objectives, such as mutual information (Tishby et al., 2000; Tishby and Zaslavsky, 2015) and rate distortion function (Chan et al., 2022). However, information measures might not effectively characterize the usefulness of features for learning tasks, due to the essentially different information processing natures. For example, originally introduced in characterizing the optimal rate of a communication system (Shannon, 1948), the mutual information is invariant to bijective transformations on variables—as the amount of information to communicate does not change after such transformations. As a consequence, when we consider the mutual information between feature representations and the input/output variables (Tishby and Zaslavsky, 2015), features up to such bijective transformations, e.g., sigmoid or exponential mappings, are all equivalent and give the same mutual information. This is in contrast to their often highly distinct performances on learning tasks. Due to this intrinsic discrepancy, it is often impractical to learn features based solely on the information measures. As a compromise, the information measures are often applied to analyze features learned from other objectives (Tishby and Zaslavsky, 2015), or used as regularization terms in the training objective to facilitate learning (Belghazi et al., 2018), where the learned features generally depend on case-by-case design choices.

In this work, we aim to establish a framework for learning feature representations that capture the statistical nature of data, and can be assembled to build different inference models without retraining. In particular, the features are learned to approximate and represent the statistical dependence of data, instead of solving specific inference tasks. To this end, we introduce a geometry on function spaces, coined *feature geometry*, which we apply to connect statistical dependence with features. This connection allows us to represent statistical dependence by corresponding operations in feature spaces. Specifically, the features are learned by approximating the statistical dependence, and the approximation error measures the amount of information carried by features. The resulting features capture the statistical dependence and thus are useful for general inference tasks.

Our main contributions of this work are as follows.

- We establish a framework for designing learning systems that separate feature learning and feature usages, where the learned features can be assembled to build different inference models without retraining. In particular, we introduce the feature geometry, which converts feature learning problems to corresponding function-space operations on statistical dependence. The resulting optimal features capture the statistical dependence of data and can be adapted to different inference tasks.

- We propose a nesting technique for decomposing statistical dependence and learning the associated feature representations. The nesting technique provides a systematic approach to construct training objectives and develop learning algorithms, where we can learn optimal features by employing the state-of-the-art deep learning practices, e.g., network architecture and optimizer designs.

- We present the applications of this unified framework in multivariate learning problems, where we design feature learning algorithms to decompose and represent the corresponding multivariate dependence. As case studies, we consider two learning scenarios: learning for conditional inference and multimodal learning with missing modalities. We investigate the optimal features and demonstrate their relations to classical learning problems, such as maximum likelihood estimation and the maximum entropy principle (Jaynes, 1957a,b).

The rest of the paper is organized as follows. In Section 2, we introduce the feature geometry, including operations on feature spaces and function representations of statistical dependence. In Section 3, we present the learning system design in a bivariate learning setting, where we demonstrate the feature learning algorithm and the design of assembling modules. In Section 4, we introduce the nesting technique, as a systematic approach to learning algorithm design in multivariate learning problems. We then demonstrate the applications of the nesting technique, where we study a conditional inference problem in Section 5, and a multimodal learning problem in Section 6. Then, we present our experimental verification for the proposed learning algorithms in Section 7, where we compare the learned features with the theoretical solutions. Finally, we summarize related works in Section 8 and provide some concluding remarks in Section 9.

2. Notations and Preliminaries

For a random variable Z, we use $\mathcal{Z}$ to denote the corresponding alphabet, and use z to denote a specific value in $\mathcal{Z}$. We use P_Z to denote the probability distribution of Z.

For the sake of simplicity, we present our development with finite alphabets and associated discrete random variables. The corresponding results can be readily extended to general alphabets under certain regularity conditions, which we discuss in Appendix A.1 for completeness.

2.1 Feature Geometry

2.1.1 Vector Space

Given an inner product space with inner product $\langle \cdot, \cdot \rangle$ and its induced norm $\| \cdot \|$, we can define the projection and orthogonal complement as follows.

Definition 1 *Give a subspace $\mathcal{W}$ of $\mathcal{V}$, we denote the projection of a vector $v \in \mathcal{V}$ onto $\mathcal{W}$ by*

$$\Pi(v; \mathcal{W}) \triangleq \arg\min_{w \in \mathcal{W}} \|v - w\|^2 . \tag{1}$$

In addition, we use $\mathcal{V} \boxminus \mathcal{W}$ to denote the orthogonal complement of $\mathcal{W}$ in $\mathcal{V}$, viz., $\mathcal{V} \boxminus \mathcal{W} \triangleq \{v \in \mathcal{V} \colon \langle v, w \rangle = 0 \text{ for all } w \in \mathcal{W}\}$.

We use "$\boxplus$" to denote the direct sum of orthogonal subspaces, i.e., $\mathcal{V} = \mathcal{V}_1 \boxplus \mathcal{V}_2$ indicates that $\mathcal{V} = \mathcal{V}_1 + \mathcal{V}_2$ and $\mathcal{V}_1 \perp \mathcal{V}_2$. Then we have the following facts.

Fact 1 *If $\mathcal{V} = \mathcal{V}_1 \boxplus \mathcal{V}_2$, then $\mathcal{V}_2 = \mathcal{V} \boxminus \mathcal{V}_1$. In addition, if $\mathcal{W}$ is a subspace of $\mathcal{V}$, then $\mathcal{V} = \mathcal{W} \boxplus (\mathcal{V} \boxminus \mathcal{W})$.*

Fact 2 (Orthogonality Principle) *Given $v \in \mathcal{V}$ and a subspace $\mathcal{W}$ of $\mathcal{V}$, then $w = \Pi(v; \mathcal{W})$ if and only if $w \in \mathcal{W}$ and $v - w \in \mathcal{V} \boxminus \mathcal{W}$. In addition, we have $v = \Pi(v; \mathcal{W}) + \Pi(v; \mathcal{V} \boxminus \mathcal{W})$.*

2.1.2 Feature Space

Given an alphabet $\mathcal{Z}$, we use $\mathcal{P}^{\mathcal{Z}}$ to denote the collection of probability distributions supported on $\mathcal{Z}$, and use $\operatorname{relint}(\mathcal{P}^{\mathcal{Z}})$ to denote the relative interior of $\mathcal{P}^{\mathcal{Z}}$, i.e., the collection of distributions $P \in \mathcal{P}^{\mathcal{Z}}$ with $P(z) > 0$ for all $z \in \mathcal{Z}$.

We use $\mathcal{F}_{\mathcal{Z}} \triangleq \{\mathcal{Z} \to \mathbb{R}\}$ to denote the collection of features (functions) on given $\mathcal{Z}$. Specifically, we use $\mathtt{1}_{\mathcal{Z}} \in \mathcal{F}_{\mathcal{Z}}$ to denote the constant feature that takes value 1, i.e., $\mathtt{1}_{\mathcal{Z}}(z) \equiv 1$ for all $z \in \mathcal{Z}$. We define the inner product on $\mathcal{F}_{\mathcal{Z}}$ as[2] $\langle f_1, f_2 \rangle \triangleq \mathbb{E}_R[f_1(Z) f_2(Z)]$, where $R \in \operatorname{relint}(\mathcal{P}^{\mathcal{Z}})$ is referred to as the **metric distribution**. This defines operations, e.g., norm and projection, on $\mathcal{F}_{\mathcal{Z}}$. Specifically, we have the induced norm $\|f\| \triangleq \sqrt{\langle f, f \rangle}$, and the projection of $f \in \mathcal{F}_{\mathcal{Z}}$ onto a subspace $\mathcal{G}_{\mathcal{Z}}$ of $\mathcal{F}_{\mathcal{Z}}$, i.e., $\Pi(f; \mathcal{G}_{\mathcal{Z}})$, is defined according to Definition 1. We also use $\tilde{\mathcal{F}}_{\mathcal{Z}} \triangleq \{f \in \mathcal{F}_{\mathcal{Z}} \colon \mathbb{E}_R[f(Z)] = 0\}$ to denote the collection of zero-mean features on $\mathcal{Z}$, which corresponds to the orthogonal complement of constant features, i.e., $\tilde{\mathcal{F}}_{\mathcal{Z}} = \mathcal{F}_{\mathcal{Z}} \boxminus \operatorname{span}\{\mathtt{1}_{\mathcal{Z}}\}$.

For each $k \geq 1$, we use $\mathcal{F}_{\mathcal{Z}}^k \triangleq (\mathcal{F}_{\mathcal{Z}})^k = \{\mathcal{Z} \to \mathbb{R}^k\}$ to denote the collection of k-dimensional features. For $f \in \mathcal{F}_{\mathcal{Z}}^k$, we use $\operatorname{span}\{f\} \subset \mathcal{F}_{\mathcal{Z}}$ to denote the subspace of $\mathcal{F}_{\mathcal{Z}}$ spanned by all dimensions of f, i.e., $\operatorname{span}\{f\} = \operatorname{span}\{f_1, \ldots, f_k\}$, and use $\Pi(f; \mathcal{G}_{\mathcal{Z}})$ to denote the corresponding projection on each dimension, i.e., $\Pi(f; \mathcal{G}_{\mathcal{Z}}) = [\Pi(f_1; \mathcal{G}_{\mathcal{Z}}), \ldots, \Pi(f_k; \mathcal{G}_{\mathcal{Z}})]^{\mathrm{T}} \in \mathcal{G}_{\mathcal{Z}}^k \triangleq (\mathcal{G}_{\mathcal{Z}})^k$. For $f_1 \in \mathcal{F}_{\mathcal{Z}}^{k_1}, f_2 \in \mathcal{F}_{\mathcal{Z}}^{k_2}$, we denote $\Lambda_{f_1, f_2} \triangleq \mathbb{E}_R\left[f_1(Z) f_2^{\mathrm{T}}(Z)\right] \in \mathbb{R}^{k_1 \times k_2}$. Specifically, we define $\Lambda_f \triangleq \Lambda_{f,f} = \mathbb{E}_R\left[f(Z) f^{\mathrm{T}}(Z)\right]$ for feature $f \in \mathcal{F}_{\mathcal{Z}}^k$. We also use $[k]$ to denote the set $\{i \in \mathbb{Z} \colon 1 \leq i \leq k\}$ with ascending order, and define $f_{[k]} \in \mathcal{F}_{\mathcal{Z}}^k$ as $f_{[k]} \colon z \mapsto (f_1(z), \ldots, f_k(z))^{\mathrm{T}}$ for given $f_1, \ldots, f_k \in \mathcal{F}_{\mathcal{Z}}$.

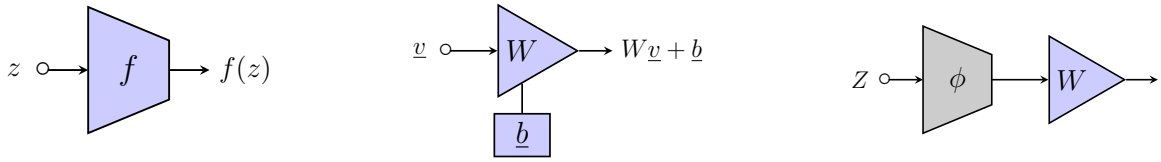
2.1.3 Neural Feature Extractors

Our development will focus on neural feature extractors, i.e., features represented by neural networks. To begin, we consider a neural network of a given architecture with m trainable parameters

2. Throughout our development, we use $\mathbb{E}_Q[f(Z)]$ to denote the expectation of function f with respect to $Q \in \mathcal{P}^{\mathcal{Z}}$, i.e., $\mathbb{E}_{\hat{Z} \sim Q}\big[f(\hat{Z})\big]$. Specifically, we will also use $\mathbb{E}[f(Z)]$ to represent $\mathbb{E}_{P_Z}[f(Z)]$, i.e., the expectation with respect to the underlying distribution; similar conventions apply to conditional expectations.

and k-dimensional outputs. Let $\mathcal{Z}$ denote the domain of input data Z, and denote the trainable parameters by[3] $\underline{\theta} \in \mathbb{R}^m$. Then, for each $\underline{\theta} \in \mathbb{R}^m$, we can denote the associated neural feature as $f^{(\underline{\theta})} \in \mathcal{F}_{\mathcal{Z}}^k$.

For the sake of simplicity, our discussions focus on the ideal case where the neural network has sufficient expressive power[4], i.e., each feature in $\mathcal{F}_{\mathcal{Z}}^k$ can be well-approximated by a large m and some $\underline{\theta} \in \mathbb{R}^m$. Under this idealized assumption, optimizing over network parameters $\underline{\theta}$ is equivalent to finding the optimal $f \in \mathcal{F}_{\mathcal{Z}}^k$. Therefore, we will omit the network parameter $\underline{\theta}$ and use f to denote such neural feature extractor, which we represented as a trapezoid block[5], as shown in Figure 2a.



(a) Feature Extractor (b) Linear Layer (c) Feature Subspace

Figure 2: Schematic representations of neural feature extractors. (a): a general feature extractor $f \in \mathcal{F}_{\mathcal{Z}}^k$; (b): a linear layer with weight matrix W and bias $\underline{b}$; (c): the composition of feature extractor blocks, where each dimension of the output lies in the feature subspace $\operatorname{span}\{\phi\}$.

The linear layer is an important building block in neural feature extractors, which performs an affine transformation on the input, as shown in Figure 2b. In particular, a linear layer with input dimension d and output dimension k is specified by a weight matrix $W \in \mathbb{R}^{k\times d}$ and bias vector $\underline{b} \in \mathbb{R}^k$, which outputs $W\underline{v} + \underline{b}$ for an input $\underline{v} \in \mathbb{R}^d$. We can use linear layers to construct feature subspaces, as shown in Figure 2c, where we have considered a given d-dimensional feature $\phi = (\phi_1, \ldots, \phi_d)^{\mathrm{T}} \in \mathcal{F}_{\mathcal{X}}^d$ and a linear layer without the bias term. Then, the collection of output features by varying weight matrix W is[6] $\{W\phi \colon W \in \mathbb{R}^{k\times d}\} = \operatorname{span}^k\{\phi\}$. Note that since the generated features are restricted to a subspace, the cascaded feature extractor has restricted expressive power, which can generally affect the feature learning. We will discuss such effects in our later developments (cf. Section 3.3).

2.1.4 Joint Functions

Given alphabets $\mathcal{X}, \mathcal{Y}$ and a metric distribution $R_{X,Y} \in \operatorname{relint}(\mathcal{P}^{\mathcal{X}\times\mathcal{Y}})$, $\mathcal{F}_{\mathcal{X}\times\mathcal{Y}}$ is composed of all joint functions of x and y. In particular, for given $f \in \mathcal{F}_{\mathcal{X}}$, $g \in \mathcal{F}_{\mathcal{Y}}$, we use $f \otimes g$ to denote their product $((x, y) \mapsto f(x) \cdot g(y)) \in \mathcal{F}_{\mathcal{X}\times\mathcal{Y}}$, and refer to such functions as *product functions*. For each product function $\gamma \in \mathcal{F}_{\mathcal{X}\times\mathcal{Y}}$, we can find $\sigma = \|\gamma\| \geq 0$, and $f \in \mathcal{F}_{\mathcal{X}}, g \in \mathcal{F}_{\mathcal{Y}}$ with $\|f\| = \|g\| = 1$, such that

$$\gamma = \sigma \cdot (f \otimes g). \tag{2}$$

We refer to (2) as the **standard form** of product functions.

3. We use underlined lowercase letters to denote Euclidean vectors, e.g., $\underline{\theta}$, $\underline{v}$, $\underline{b}$, to distinguish them from scalars.
4. Such idealized network can be constructed by using the universal approximation theorem; see, e.g.,Cybenko (1989), for detailed discussions.
5. The literature sometimes use the longer (shorter) bases to indicate the larger (smaller) dimensions of input and output. However, our schematic representation does not make this distinction.
6. If we also consider the bias term $\underline{b}$, the feature subspace will become $\operatorname{span}^k\{\mathtt{1}_{\mathcal{Z}}, \phi_1, \ldots, \phi_d\}$, where $\mathtt{1}_{\mathcal{Z}}$ represents the constant function $\mathcal{Z} \ni z \mapsto 1$.

In addition, for given $f = (f_1, \ldots, f_k)^{\mathrm{T}} \in \mathcal{F}_{\mathcal{X}}^k$ and $g = (g_1, \ldots, g_k)^{\mathrm{T}} \in \mathcal{F}_{\mathcal{Y}}^k$, we denote $f \otimes g \triangleq \sum_{i=1}^k f_i \otimes g_i$. For two subspaces $\mathcal{G}_{\mathcal{X}}$ and $\mathcal{G}_{\mathcal{Y}}$ of $\mathcal{F}_{\mathcal{X}}$ and $\mathcal{F}_{\mathcal{Y}}$, respectively, we denote the tensor product of $\mathcal{G}_{\mathcal{X}}$ and $\mathcal{G}_{\mathcal{Y}}$ as $\mathcal{G}_{\mathcal{X}} \otimes \mathcal{G}_{\mathcal{Y}} \triangleq \operatorname{span}\{f \otimes g \colon f \in \mathcal{G}_{\mathcal{X}}, g \in \mathcal{G}_{\mathcal{Y}}\}$.

Note that by extending each $f = (x \mapsto f(x)) \in \mathcal{F}_{\mathcal{X}}$ to $((x, y) \mapsto f(x)) \in \mathcal{F}_{\mathcal{X}\times\mathcal{Y}}$, $\mathcal{F}_{\mathcal{X}}$ becomes a subspace of $\mathcal{F}_{\mathcal{X}\times\mathcal{Y}}$, with the metric distribution being the marginal distribution R_X of $R_{X,Y}$. We then denote the orthogonal complement of $\mathcal{F}_{\mathcal{X}}$ in $\mathcal{F}_{\mathcal{X}\times\mathcal{Y}}$ as

$$\mathcal{F}_{\mathcal{Y}|\mathcal{X}} \triangleq \mathcal{F}_{\mathcal{X}\times\mathcal{Y}} \boxminus \mathcal{F}_{\mathcal{X}}. \tag{3}$$

We establish a correspondence between the distribution space $\mathcal{P}^{\mathcal{Z}}$ and the feature space $\mathcal{F}_{\mathcal{Z}}$ by the density ratio function.

Definition 2 *Given a metric distribution $R \in \operatorname{relint}(\mathcal{P}^{\mathcal{Z}})$, for each $P \in \mathcal{P}^{\mathcal{Z}}$, we define the (centered) density ratio function $\tilde{\ell}_{P;R} \in \mathcal{F}_{\mathcal{Z}}$ as*

$$\tilde{\ell}_{P;R}(z) \triangleq \frac{P(z) - R(z)}{R(z)}, \quad \textit{for all } z \in \mathcal{Z}.$$

It is easy to verify that $\tilde{\ell}_{P;R}$ has mean zero , i.e., $\tilde{\ell}_{P;R} \in \tilde{\mathcal{F}}_{\mathcal{Z}}$. We will simply refer to $\tilde{\ell}_{P;R}$ as the density ratio or likelihood ratio and use $\tilde{\ell}_P$ to denote $\tilde{\ell}_{P;R}$ when there is no ambiguity about the metric distribution R.

In particular, given random variables X and Y with the joint distribution $P_{X,Y}$, we denote the density ratio $\tilde{\ell}_{P_{X,Y};P_XP_Y}$ by $\mathfrak{i}_{X;Y}$, i.e.,

$$\mathfrak{i}_{X;Y}(x, y) = \frac{P_{X,Y}(x, y) - P_X(x)P_Y(y)}{P_X(x)P_Y(y)}, \quad \text{for all } x \in \mathcal{X}, y \in \mathcal{Y}. \tag{4}$$

We refer to $\mathfrak{i}_{X;Y}$ as the **canonical dependence kernel** (CDK) function, which connects the $(X;Y)$ dependence with $\mathcal{F}_{\mathcal{X}\times\mathcal{Y}}$.

With the feature geometry, we can associate function-space concepts with corresponding operations on features, which we summarize as follows.

Property 1 *Consider the feature geometry on $\mathcal{F}_{\mathcal{X}\times\mathcal{Y}}$ defined by the metric distribution $R_{X,Y} = P_XP_Y$. Then, we have $\langle f_1 \otimes g_1, f_2 \otimes g_2\rangle = \mathbb{E}_{P_X}[f_1(X)f_2(X)] \cdot \mathbb{E}_{P_Y}[g_1(Y)g_2(Y)]$ for given $f_1, f_2 \in \mathcal{F}_{\mathcal{X}}, g_1, g_2 \in \mathcal{F}_{\mathcal{Y}}$. In addition, For any $k \geq 1$ and $f \in \mathcal{F}_{\mathcal{X}}^k, g \in \mathcal{F}_{\mathcal{Y}}^k$, we have*

$$\Pi(f; \operatorname{span}\{\mathtt{1}_{\mathcal{X}}\}) = \mathbb{E}_{P_X}[f(X)], \quad \Pi\left(f; \tilde{\mathcal{F}}_{\mathcal{X}}\right) = f - \mathbb{E}_{P_X}[f(X)], \tag{5}$$

$$\langle \mathfrak{i}_{X;Y}, f \otimes g\rangle = \mathbb{E}_{P_{X,Y}}\left[f^{\mathrm{T}}(X)g(Y)\right] - \left(\mathbb{E}_{P_X}[f(X)]\right)^{\mathrm{T}} \mathbb{E}_{P_Y}[g(Y)], \tag{6}$$

$$\|f \otimes g\|^2 = \operatorname{tr}(\Lambda_f\Lambda_g), \tag{7}$$

where $\Lambda_f = \mathbb{E}_{P_X}\left[f(X)f^{\mathrm{T}}(X)\right], \Lambda_g = \mathbb{E}_{P_Y}\left[g(Y)g^{\mathrm{T}}(Y)\right]$.

2.1.5 Feature Geometry on Data Samples

In practice, the variables of interest typically have unknown and complicated probability distributions, with only data samples available for learning. We can similarly define the feature geometry on data samples by considering the corresponding empirical distributions.

To begin, given a dataset consisting of n samples of Z, represented as $\{z_i\}_{i=1}^n$, we denote the corresponding empirical distribution $\hat{P}_Z \in \mathcal{P}^{\mathcal{Z}}$ by

$$\hat{P}_Z(z) \triangleq \frac{1}{n}\sum_{i=1}^n \mathbb{1}_{\{z_i = z\}}, \quad \text{for all } z \in \mathcal{Z}, \tag{8}$$

where $\mathbb{1}_{\{\cdot\}}$ denotes the indicator function. Then, for any function f of Z, we have $\mathbb{E}_{\hat{P}_Z}[f(Z)] = \sum_{z\in\mathcal{Z}} \hat{P}_Z(z)\cdot f(z) = \frac{1}{n}\sum_{i=1}^n f(z_i)$, which is the empirical average of f over the dataset.

Specifically, given a dataset $\mathscr{D} = \{(x_i, y_i)\}_{i=1}^n$ with n sample pairs of (X, Y), the corresponding joint empirical distribution $\hat{P}_{X,Y} \in \mathcal{P}^{\mathcal{X}\times\mathcal{Y}}$ is given by

$$\hat{P}_{X,Y}(x,y) = \frac{1}{n}\sum_{i=1}^n \mathbb{1}_{\{x_i=x, y_i=y\}}, \quad \text{for all } x\in\mathcal{X}, y\in\mathcal{Y}. \tag{9}$$

It is easily verified that the marginal distributions of $\hat{P}_{X,Y}$ are the empirical distributions $\hat{P}_X$ of $\{x_i\}_{i=1}^n$ and $\hat{P}_Y$ of $\{y_i\}_{i=1}^n$. Therefore, we can express the CDK function associated with the empirical distribution $\hat{P}_{X,Y}$ as [cf. (4)]

$$\hat{\mathfrak{i}}_{X;Y}(x,y) = \frac{\hat{P}_{X,Y}(x,y) - \hat{P}_X(x)\hat{P}_Y(y)}{\hat{P}_X(x)\hat{P}_Y(y)}, \quad \text{for all } x\in\mathcal{X}, y\in\mathcal{Y}. \tag{10}$$

As a result, given the dataset $\mathscr{D}$, we can define the associated feature geometry on $\mathcal{F}_{\mathcal{X}\times\mathcal{Y}}$ by using the metric distribution $R_{X,Y} = \hat{P}_X\hat{P}_Y$. From Property 1, we can evaluate the induced geometric quantities over data samples in $\mathscr{D}$, by replacing the distributions by the corresponding empirical distributions, and applying the empirical CDK function $\hat{\mathfrak{i}}_{X;Y}$ in (6).

Such characteristic allows us to discuss theoretical properties and algorithmic implementations in a unified notation. In our later developments, we will state both theoretical results and algorithms designs using the same notation of distribution, say, $P_{X,Y}$. This $P_{X,Y}$ corresponds to the underlying distribution in theoretical analyses, and represents the corresponding empirical distribution in algorithm implementations.

Remark 3 *Note that for finite $\mathcal{Z}$, $\mathcal{F}_\mathcal{Z}$ is a space with a finite dimension $|\mathcal{Z}|$. Therefore, we can choose a basis of $\mathcal{F}_\mathcal{Z}$ and represent each feature as corresponding coefficient vectors in Euclidean space $\mathbb{R}^{|\mathcal{Z}|}$. Similarly, we can represent operators on function spaces as matrices. Such conventions have been introduced and adopted in previous works, e.g., (Huang et al., 2024; Xu et al., 2022), which we summarize in Appendix A.2 for completeness and comparisons.*

2.2 Modal Decomposition

We then investigate a representation of joint functions in $\mathcal{F}_{\mathcal{X}\times\mathcal{Y}}$ by using features in $\mathcal{F}_\mathcal{X}$ and $\mathcal{F}_\mathcal{Y}$. This representation will be our basic tool to connect statistical dependence with feature spaces. Throughout this section, we set the metric distribution of $\mathcal{F}_{\mathcal{X}\times\mathcal{Y}}$ to the product form[7], i.e., $R_{X,Y} = R_X R_Y$. This induced metric distributions on $\mathcal{F}_\mathcal{X}$ and $\mathcal{F}_\mathcal{Y}$ are R_X and R_Y, respectively.

2.2.1 Definitions and Properties

We use the operator ζ on $\mathcal{F}_{\mathcal{X}\times\mathcal{Y}}$ to denote the optimal rank-1 approximation, i.e.,

$$\zeta(\gamma) \triangleq \operatorname*{arg\,min}_{\substack{\gamma' :\, \gamma' = f\otimes g \\ f\in\mathcal{F}_\mathcal{X}, g\in\mathcal{F}_\mathcal{Y}}} \|\gamma - \gamma'\|, \quad \text{for all } \gamma\in\mathcal{F}_{\mathcal{X}\times\mathcal{Y}}. \tag{11}$$

In addition, for all $k \geq 1$, we define the operator ζ_k as $\zeta_1 \triangleq \zeta$ and

$$\zeta_k(\gamma) \triangleq \zeta\left(\gamma - \sum_{i=1}^{k-1}\zeta_i(\gamma)\right), \tag{12}$$

7. Under this choice, the norm on $\mathcal{F}_{\mathcal{X}\times\mathcal{Y}}$ corresponds to the Frobenius (Hilbert–Schmidt) norm.

which we refer to as the k-th mode of γ. Then, we use

$$\zeta_{\leq k}(\gamma) \triangleq \sum_{i=1}^{k} \zeta_i(\gamma) \qquad \text{and} \qquad r_k(\gamma) \triangleq \gamma - \zeta_{\leq k}(\gamma) \tag{13}$$

to denote the superposition of the top k modes and the corresponding remainder, respectively.

Remark 4 *If the minimization problem* (11) *has multiple solutions, the corresponding $\zeta(\gamma)$ (and $\zeta_k(\gamma)$) might not be unique. In this case, $\zeta_1(\gamma), \dots \zeta_k(\gamma)$ represent one of such solutions obtained from the sequential rank-1 approximations.*

For each $\gamma \in \mathcal{F}_{\mathcal{X}\times\mathcal{Y}}$, we define the rank of γ as $\operatorname{rank}(\gamma) \triangleq \inf\{k \geq 0 \colon \|r_k(\gamma)\| = 0\}$. Suppose $K \triangleq \operatorname{rank}(\gamma)$, then from the definition (13), we have $\gamma = \zeta_{\leq K}(\gamma) + r_K(\gamma) = \zeta_{\leq K}(\gamma)$, i.e.,

$$\gamma = \sum_{i=1}^{K} \zeta_i(\gamma). \tag{14}$$

For each $i \in [K]$, let us denote the standard form [cf. (2)] of $\zeta_i(\gamma)$ by $\zeta_i(\gamma) = \sigma_i (f_i^* \otimes g_i^*)$. Therefore, from (14) we obtain

$$\gamma(x, y) = \sum_{i=1}^{K} \sigma_i \cdot f_i^*(x) \cdot g_i^*(y), \quad \text{for all } x \in \mathcal{X},\, y \in \mathcal{Y}, \tag{15}$$

where $\|f_i^*\| = \|g_i^*\| = 1$ and $\sigma_i = \|\zeta_i(\gamma)\|$. We refer to (15) as the *modal decomposition* of γ, which is a special case of Schmidt decomposition (Schmidt, 1907; Ekert and Knight, 1995), or singular value decomposition (SVD) in function space. We list several useful characterizations as follows.

Fact 3 *Let $K \triangleq \operatorname{rank}(\gamma)$, then $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_K > 0$. In addition, for all $i, j = 1, \dots, K$, we have*[8] *$\langle f_i^*, f_j^* \rangle = \langle g_i^*, g_j^* \rangle = \delta_{ij}$, and*

$$\begin{aligned} \langle \zeta_i(\gamma), \zeta_j(\gamma) \rangle &= 0, \qquad \text{if } i < j, \\ \langle \zeta_i(\gamma), r_j(\gamma) \rangle &= 0, \qquad \text{if } i \leq j, \end{aligned}$$

where $r_j(\cdot)$ is as defined in (13).

Fact 4 *For all $i \in [K]$, we have $(f_i^*, g_i^*) = \operatorname*{arg\,max}_{(f,g)\in\mathcal{D}_i} \langle \gamma, f \otimes g \rangle$ where we have recursively defined each $\mathcal{D}_i$ as $\mathcal{D}_i = \{(f, g) \in \mathcal{F}_{\mathcal{X}} \times \mathcal{F}_{\mathcal{Y}} \colon \|f\| = \|g\| = 1$ and $\langle f, f_j^* \rangle = \langle g, g_j^* \rangle = 0$ for all $j \in [i-1]\}$.*

Fact 5 (Eckart–Young–Mirsky theorem, Eckart and Young 1936) *For all $\gamma \in \mathcal{F}_{\mathcal{X}\times\mathcal{Y}}$ and $k \geq 1$, we have*

$$\zeta_{\leq k}(\gamma) = \operatorname*{arg\,min}_{\gamma' \colon \operatorname{rank}(\gamma') \leq k} \|\gamma - \gamma'\| = \operatorname*{arg\,min}_{\substack{\gamma' \colon \gamma' = f \otimes g, \\ f \in \mathcal{F}_{\mathcal{X}}^k,\, g \in \mathcal{F}_{\mathcal{Y}}^k}} \|\gamma - \gamma'\|.$$

Therefore, we refer to $\zeta_{\leq k}(\gamma)$ as the rank-k approximation of γ, and the remainder $r_k(\gamma)$ represents the approximation error.

8. We adopt the Kronecker delta notation

$$\delta_{ij} = \begin{cases} 0 & \text{if } i \neq j, \\ 1 & \text{if } i = j. \end{cases}$$

2.2.2 Constrained Modal Decomposition

We then introduce the constrained modal decomposition, which provides an effective implementation of projection operators. For given subspace $\mathcal{G}_\mathcal{X}$ of $\mathcal{F}_\mathcal{X}$ and subspace $\mathcal{G}_\mathcal{Y}$ of $\mathcal{F}_\mathcal{Y}$, we define

$$\zeta(\gamma|\mathcal{G}_\mathcal{X}, \mathcal{G}_\mathcal{Y}) \triangleq \operatorname*{arg\,min}_{\substack{\gamma' :\, \gamma' = f \otimes g \\ f \in \mathcal{G}_\mathcal{X}, g \in \mathcal{G}_\mathcal{Y}}} \|\gamma - \gamma'\|, \tag{16}$$

$$\zeta_k(\gamma|\mathcal{G}_\mathcal{X}, \mathcal{G}_\mathcal{Y}) \triangleq \zeta\left(\gamma - \sum_{i=1}^{k-1} \zeta_i(\gamma|\mathcal{G}_\mathcal{X}, \mathcal{G}_\mathcal{Y}) \middle| \mathcal{G}_\mathcal{X}, \mathcal{G}_\mathcal{Y}\right), \quad \text{for all } k \geq 1, \tag{17}$$

where $\zeta_1(\gamma|\mathcal{G}_\mathcal{X}, \mathcal{G}_\mathcal{Y}) \triangleq \zeta(\gamma|\mathcal{G}_\mathcal{X}, \mathcal{G}_\mathcal{Y})$. Similarly, we denote $\zeta_{\leq k}(\gamma|\mathcal{G}_\mathcal{X}, \mathcal{G}_\mathcal{Y}) \triangleq \sum_{i=1}^{k} \zeta_i(\gamma|\mathcal{G}_\mathcal{X}, \mathcal{G}_\mathcal{Y})$.

We can extend the properties of modal decomposition to the constrained case. In particular, we have the following extension of Fact 5, of which a proof is provided in Appendix C.1.

Proposition 5 *Suppose $\mathcal{G}_\mathcal{X}$ and $\mathcal{G}_\mathcal{Y}$ are subspace of $\mathcal{F}_\mathcal{X}$ and $\mathcal{F}_\mathcal{Y}$, respectively. Then, for all $\gamma \in \mathcal{F}_{\mathcal{X}\times\mathcal{Y}}$ and $k \geq 1$, we have $\zeta_k(\gamma|\mathcal{G}_\mathcal{X}, \mathcal{G}_\mathcal{Y}) = \zeta_k(\Pi(\gamma; \mathcal{G}_\mathcal{X} \otimes \mathcal{G}_\mathcal{Y}))$, and*

$$\zeta_{\leq k}(\gamma|\mathcal{G}_\mathcal{X}, \mathcal{G}_\mathcal{Y}) = \zeta_{\leq k}(\Pi(\gamma; \mathcal{G}_\mathcal{X} \otimes \mathcal{G}_\mathcal{Y})) = \operatorname*{arg\,min}_{\substack{\gamma' :\, \gamma' = f \otimes g, \\ f \in \mathcal{G}_\mathcal{X}^k, g \in \mathcal{G}_\mathcal{Y}^k}} \|\gamma - \gamma'\|, \tag{18}$$

where we have defined $\mathcal{G}_\mathcal{X}^k \triangleq (\mathcal{G}_\mathcal{X})^k$ and $\mathcal{G}_\mathcal{Y}^k \triangleq (\mathcal{G}_\mathcal{Y})^k$.

Therefore, we can implement projection operators by solving an equivalent constrained low-rank approximation (or modal decomposition) problem.

2.3 Statistical Dependence and Induced Features

Given $(X, Y) \sim P_{X,Y}$, we consider the space $\mathcal{F}_{\mathcal{X}\times\mathcal{Y}}$ with the metric distribution $R_{X,Y} = P_X P_Y$. Then, we can characterize the statistical dependence between X and Y by the CDK function $\mathfrak{i}_{X;Y}$, as defined in (4). We can characterize the energy of $\mathfrak{i}_{X;Y}$ as $\|\mathfrak{i}_{X;Y}\|^2$, also known as the *mean square contingency* (Pearson, 1904). Specifically, it is easy to verify that $\|\mathfrak{i}_{X;Y}\| = 0$ if and only if X and Y are independent.

Suppose $\operatorname{rank}(\mathfrak{i}_{X;Y}) = K$ and let the modal decomposition be [cf. (15)]

$$\mathfrak{i}_{X;Y} = \sum_{i=1}^{K} \zeta_i(\mathfrak{i}_{X;Y}) = \sum_{i=1}^{K} \sigma_i \cdot (f_i^* \otimes g_i^*), \tag{19}$$

where for each $i \in [K]$, $\zeta_i(\mathfrak{i}_{X;Y}) = \sigma_i \cdot (f_i^* \otimes g_i^*)$ is the standard form of i-th rank-one dependence mode, with strength characterized by $\sigma_i = \|\zeta_i(\mathfrak{i}_{X;Y})\|$. Note that since different modes are orthogonal (cf. Fact 3), we have $\left\|\mathfrak{i}_{X;Y}\right\|^2 = \sum_{i=1}^{K} \sigma_i^2$. From $\sigma_1 \geq \cdots \geq \sigma_K$, these modes are ordered by their contributions to the joint dependence.

In particular, the features f_i^*'s, g_i^*'s are the maximally correlated features in $\mathcal{F}_\mathcal{X}$, $\mathcal{F}_\mathcal{Y}$, known as Hirschfeld–Gebelein–Rényi (HGR) maximal correlation functions (Hirschfeld, 1935; Gebelein, 1941; Rényi, 1959). To see this, let us denote the covariance for given $f \in \mathcal{F}_\mathcal{X}, g \in \mathcal{F}_\mathcal{Y}$ as

$$\operatorname{cov}(f, g) \triangleq \mathbb{E}_{P_{X,Y}}[f(X)g(Y)] - \mathbb{E}_{P_X P_Y}[f(X)g(Y)]. \tag{20}$$

From Fact 4 and the fact $\operatorname{cov}(f, g) = \langle \mathfrak{i}_{X;Y}, f \otimes g \rangle$, we obtain the following corollary.

Corollary 6 (HGR Maximal Correlation Functions) *For each $i = 1, \dots, K$, we have $\sigma_i = \operatorname{cov}(f_i^*, g_i^*) = \mathbb{E}_{P_{X,Y}}[f_i^*(X) g_i^*(Y)]$ and $(f_i^*, g_i^*) = \arg\max_{(f,g)\in\mathcal{D}_i} \operatorname{cov}(f_i, g_i)$, where we have recursively defined each $\mathcal{D}_i$ as $\mathcal{D}_i = \{(f,g) \in \mathcal{F}_\mathcal{X} \times \mathcal{F}_\mathcal{Y} \colon \|f\| = \|g\| = 1 \text{ and } \langle f, f_j^* \rangle = \langle g, g_j^* \rangle = 0 \text{ for all } j \in [i-1]\}$.*

We can further extend the results to the constrained modal decomposition (cf. Section 2.2.2) of $\mathfrak{i}_{X;Y}$. Specifically, given subspaces $\mathcal{G}_\mathcal{X}$ and $\mathcal{G}_\mathcal{Y}$ of $\mathcal{F}_\mathcal{X}$ and $\mathcal{F}_\mathcal{Y}$, respectively, let

$$\zeta_i(\mathfrak{i}_{X;Y} | \mathcal{G}_\mathcal{X}, \mathcal{G}_\mathcal{Y}) = \hat{\sigma}_i \cdot (\hat{f}_i^* \otimes \hat{g}_i^*), \quad i \geq 1, \tag{21}$$

be corresponding standard form representations. Then, we can interpret $\hat{\sigma}_i, \hat{f}_i^*, \hat{g}_i^*$ as the solution to a constrained maximal correlation problem, formalized as the following extension of Corollary 6. A proof is provided in Appendix C.2.

Proposition 7 *Given subspaces $\mathcal{G}_\mathcal{X}$ and $\mathcal{G}_\mathcal{Y}$ of $\mathcal{F}_\mathcal{X}$ and $\mathcal{F}_\mathcal{Y}$, respectively, for each $i \geq 1$, we have $\hat{\sigma}_i = \operatorname{cov}(\hat{f}_i^*, \hat{g}_i^*) = \mathbb{E}_{P_{X,Y}}\big[\hat{f}_i^*(X)\hat{g}_i^*(Y)\big]$, $(\hat{f}_i^*, \hat{g}_i^*) = \arg\max_{(f,g)\in\hat{\mathcal{D}}_i} \operatorname{cov}(f,g)$, where* cov *denotes the covariance [cf. (20)], and where we have recursively defined each $\hat{\mathcal{D}}_i$ as $\hat{\mathcal{D}}_i = \{(f,g) \in \mathcal{G}_\mathcal{X} \times \mathcal{G}_\mathcal{Y} \colon \|f\| = \|g\| = 1 \text{ and } \langle f, \hat{f}_j^* \rangle = \langle g, \hat{g}_j^* \rangle = 0 \text{ for all } j \in [i-1]\}$.*

In particular, we can interpret CCA (Canonical Correlation Analysis) as the modal decomposition constrained to linear functions.

Example 1 (Canonical Correlation Analysis) *Suppose $\mathcal{X}$ and $\mathcal{Y}$ are vector spaces, and $\mathcal{G}_\mathcal{X}, \mathcal{G}_\mathcal{Y}$ are the space of all linear functions defined on $\mathcal{X}$, $\mathcal{Y}$, respectively. Then, Proposition 7 gives solutions to CCA (Canonical Correlation Analysis) (Hotelling, 1936), where $\hat{\sigma}_i$'s are canonical correlations.*

Weak Dependence and Local Geometric Analyses In the particular case where the statistical dependence between X and Y is weak, we can establish further connections between feature geometry and conventional information measures. Such analyses have been extensively studied in Huang et al. (2024), referred to as the local geometric analysis, formalized as follows.

Definition 8 (ϵ-Dependence) *Given $(X,Y) \sim P_{X,Y}$, X and Y are ϵ-dependent if $\|\mathfrak{i}_{X;Y}\| = O(\epsilon)$.*

For such weakly dependent variables, we can characterize their mutual information as follows.

Lemma 9 (Huang et al. 2024, Lemma 4.11) *If X and Y are ϵ-dependent, then we have the mutual information $I(X;Y) = \frac{1}{2} \cdot \left\|\mathfrak{i}_{X;Y}\right\|^2 + o(\epsilon^2)$.*

Therefore, from (19) we can also decompose the mutual information into different modes: $I(X;Y) = \frac{1}{2} \cdot \left\|\mathfrak{i}_{X;Y}\right\|^2 + o(\epsilon^2) = \frac{1}{2} \sum_{i=1}^K \sigma_i^2 + o(\epsilon^2)$.

3. Dependence Approximation and Feature Learning

In this section, we demonstrate the learning system design with feature geometry in a learning setting. In particular, we consider optimal feature representations of the statistical dependence, and present learning such features from data and assembling them to build inference models. To begin, let X and Y denote the random variables of interest, with the joint distribution $P_{X,Y}$. We characterize the statistical dependence between X and Y as the CDK function [cf. (4)] $\mathfrak{i}_{X;Y} \in \mathcal{F}_{\mathcal{X}\times\mathcal{Y}}$. In our development, we consider the feature geometry on $\mathcal{F}_{\mathcal{X}\times\mathcal{Y}}$ with respect to the metric distribution $R_{X,Y} = P_X P_Y$. We also assume $\mathfrak{i}_{X;Y}$ has the modal decomposition (19).

3.1 Low Rank Approximation of Statistical Dependence

In learning applications, the joint distribution $P_{X,Y}$ is typically unknown with enormous complexity, making direct computation or estimation of $\mathfrak{i}_{X;Y}$ infeasible. To tackle this problem, we consider the representation of $\mathfrak{i}_{X;Y}$ using features of X and Y, and develop the feature learning algorithms that can be effectively implemented on (X, Y) samples.

Specifically, for given $k \geq 1$ and k-dimensional features $f \in \mathcal{F}_{\mathcal{X}}^k$ and $g \in \mathcal{F}_{\mathcal{Y}}^k$, we consider the approximation of $\mathfrak{i}_{X;Y}$ by the rank-k joint function $f \otimes g = \sum_{i=1}^k f_i \otimes g_i$. With this formulation, we can convert the computation of the rank-k approximation $\zeta_{\leq k}(\mathfrak{i}_{X;Y})$ to an optimization problem, where the objective is the approximation error $\|\mathfrak{i}_{X;Y} - f \otimes g\|$, and where the optimization variables are k-dimensional features f and g. Then, $\zeta_{\leq k}(\mathfrak{i}_{X;Y})$ can be represented by the resulting optimal features in a factorized form.

However, we cannot directly compute the error $\|\mathfrak{i}_{X;Y} - f \otimes g\|$ for given f and g, due to the unknown $\mathfrak{i}_{X;Y}$. To address this issue, we introduce the H-score, proposed in (Xu and Huang, 2020; Xu et al., 2022).

Definition 10 *Given $k \geq 1$ and $f \in \mathcal{F}_{\mathcal{X}}^k$, $g \in \mathcal{F}_{\mathcal{Y}}^k$, the H-score $\mathscr{H}(f, g)$ is defined as*

$$\mathscr{H}(f,g) \triangleq \frac{1}{2}\left(\left\|\mathfrak{i}_{X;Y}\right\|^2 - \|\mathfrak{i}_{X;Y} - f \otimes g\|^2\right) \tag{22}$$

$$= \mathbb{E}\left[f^{\mathrm{T}}(X)g(Y)\right] - \left(\mathbb{E}\left[f(X)\right]\right)^{\mathrm{T}} \mathbb{E}\left[g(Y)\right] - \frac{1}{2} \cdot \mathrm{tr}\left(\Lambda_f \Lambda_g\right), \tag{23}$$

where $\Lambda_f = \mathbb{E}\left[f(X)f^{\mathrm{T}}(X)\right]$, $\Lambda_g = \mathbb{E}\left[g(Y)g^{\mathrm{T}}(Y)\right]$.

The H-score measures the goodness of the approximation, with a larger H-score value indicating a smaller approximation error. In particular, for k-dimensional feature inputs, the maximum value of H-score gives the total energy of top-k dependence modes, achieved by the optimal rank-k approximation. Formally, we have the following property from Fact 5.

.

Property 2 *Given $k \geq 1$, let $\sigma_i = \|\zeta_i(\mathfrak{i}_{X;Y})\|$ for $i \in [k]$. Then, for all $f \in \mathcal{F}_{\mathcal{X}}^k$ and $g \in \mathcal{F}_{\mathcal{Y}}^k$,*

$$\mathscr{H}(f,g) \leq \frac{1}{2}\|\zeta_{\leq k}(\mathfrak{i}_{X;Y})\|^2 = \frac{1}{2}\sum_{i=1}^k \sigma_i^2, \tag{24}$$

where the inequality holds with equality if and only if $f \otimes g = \zeta_{\leq k}(\mathfrak{i}_{X;Y})$.

In practice, for given features f and g, we can efficiently compute the H-score $\mathscr{H}(f, g)$ from data samples, by evaluating corresponding empirical averages in (23). Since $\mathscr{H}(f, g)$ is differentiable with respect to f and g, we can use it as the training objective for learning the low-rank approximation of $\mathfrak{i}_{X;Y}$, where we use neural networks to parameterize f and g and optimize their parameters by batch (minibatch) gradient descent. Suppose the networks have sufficient expressive power, then the optimal solution gives the desired low-rank approximation $\zeta_{\leq k}(\mathfrak{i}_{X;Y})$.

Remark 11 *It is worth mentioning that the optimal features learned from finite data samples correspond to the modal decomposition of the associated empirical distribution (cf. Section 2.1.5), which is generally different from the underlying distribution. As a result, the learned features will deviate from the theoretical values of features; see, e.g., Huang and Xu (2020); Makur et al. (2020) for detailed discussions on the sample complexity of learning such features.*

Note that in this particular bivariate setting, the roles of X and Y (and the learned features f and g) are symmetric. Moreover, we learn the features by directly factorizing the statistical dependence between X and Y, instead of solving a specific inference task, e.g., predicting Y based on X, or vice versa. Nevertheless, we can readily solve these inference tasks by simply assembling the learned features, as we will demonstrate next.

3.2 Feature Assembling and Inference Models

With features $f \in \mathcal{F}_{\mathcal{X}}^k, g \in \mathcal{F}_{\mathcal{Y}}^k$ learned from maximizing the H-score $\mathscr{H}(f,g)$, we then discuss the construction of different inference models by assembling these features. We first consider the case where $k \geq \operatorname{rank}(\mathfrak{i}_{X;Y})$ and we have learned $f \otimes g = \mathfrak{i}_{X;Y}$ (cf. Property 2). Then, we have the following proposition. A proof is provided in Appendix C.3.

Proposition 12 *Suppose $f \otimes g = \mathfrak{i}_{X;Y}$. Then, we have $\|\mathfrak{i}_{X;Y}\|^2 = \operatorname{tr}(\Lambda_f \Lambda_g)$ and*

$$P_{Y|X}(y|x) = P_Y(y)\left(1 + f^{\mathrm{T}}(x)g(y)\right). \tag{25}$$

In addition, for any d-dimensional function $\psi \in \mathcal{F}_{\mathcal{Y}}^d$, we have

$$\mathbb{E}\left[\psi(Y)|X=x\right] = \mathbb{E}\left[\psi(Y)\right] + \Lambda_{\psi,g} f(x), \tag{26}$$

where $\Lambda_{\psi,g} = \mathbb{E}\left[\psi(Y)g^{\mathrm{T}}(Y)\right]$.

Therefore, we can compute the strength of $(X;Y)$ dependence, i.e., $\|\mathfrak{i}_{X;Y}\|$ from the features f and g. In addition, the posterior distribution (25) and conditional expectation (26) are useful for supervised learning tasks. Specifically, we consider the case where X and Y are the input variable and target variable, respectively. The Y represents the categorical label in classification tasks, or the target to estimate in regression tasks.

In classification tasks, we can compute the posterior distribution $P_{Y|X}$ of the label Y from (25). The corresponding corresponding MAP (maximum a posteriori) estimation is

$$\hat{y}_{\mathrm{MAP}}(x) = \arg\max_{y \in \mathcal{Y}} P_{Y|X}(y|x) = \arg\max_{y \in \mathcal{Y}} P_Y(y)\left(1 + f^{\mathrm{T}}(x)g(y)\right), \tag{27}$$

where P_Y can be obtained from training set. This approach is also referred to as the maximal correlation regression (MCR) (Xu and Huang, 2020). Similarly, the maximum likelihood estimation (MLE) is given by

$$\hat{y}_{\mathrm{MLE}}(x) = \arg\max_{y \in \mathcal{Y}} P_{X|Y}(x|y) = \arg\max_{y \in \mathcal{Y}} f^{\mathrm{T}}(x)g(y). \tag{28}$$

If the target variable Y is continuous, it is often of interest to estimate Y, or more generally, some transformation ψ of Y. Then, the MMSE (minimum mean square error) estimation of $\psi(Y)$ based on $X = x$ is the conditional expectation $\mathbb{E}\left[\psi(Y)|X=x\right]$. From (26), we can efficiently compute the conditional expectation, where $\mathbb{E}\left[\psi(Y)\right]$ and $\Lambda_{\psi,g} = \mathbb{E}\left[\psi(Y)g^{\mathrm{T}}(Y)\right]$ can be evaluated from the training dataset by taking the corresponding empirical averages. Therefore, we obtain the model for estimating $\psi(Y)$ for any given ψ, by simply assembling the learned features without retraining.

In practice, it can happen that feature dimension $k < \operatorname{rank}(\mathfrak{i}_{X;Y})$, due to a potentially large $\operatorname{rank}(\mathfrak{i}_{X;Y})$. In such case, the best approximation of $\mathfrak{i}_{X;Y}$ would be the rank-k approximation $\zeta_{\leq k}(\mathfrak{i}_{X;Y})$, and we can establish a similar result as follows. A proof is provided in Appendix C.4.

Proposition 13 *Suppose $f \otimes g = \zeta_{\le k}(\mathfrak{i}_{X;Y})$ for $k \ge 1$. Then, for all d-dimensional function $\psi \in \operatorname{span}^d\{\mathtt{1}_{\mathcal{Y}}, g_1^*, \dots, g_k^*\}$, we have $\mathbb{E}[\psi(Y)|X=x] = \mathbb{E}[\psi(Y)] + \Lambda_{\psi,g} f(x)$, where $\mathtt{1}_{\mathcal{Y}}$ is the constant function $(y \mapsto 1) \in \mathcal{F}_{\mathcal{Y}}$, and where for each $i \in [k]$, g_i^* is obtained from the standard form of $\zeta_i(\mathfrak{i}_{X;Y})$: $\sigma_i(f_i^* \otimes g_i^*) = \zeta_i(\mathfrak{i}_{X;Y})$.*

3.3 Constrained Dependence Approximation

We can readily extend the above analysis to the constrained low-rank approximation problem. Specifically, we consider the constrained rank-k approximation [cf. (18)] $\zeta_{\le k}(\mathfrak{i}_{X;Y}|\mathcal{G}_{\mathcal{X}}, \mathcal{G}_{\mathcal{Y}})$ for $k \ge 1$, where $\mathcal{G}_{\mathcal{X}}$ and $\mathcal{G}_{\mathcal{Y}}$ are subspaces of $\mathcal{F}_{\mathcal{X}}$ and $\mathcal{F}_{\mathcal{Y}}$, respectively. Analogous to Property 2, when we restrict $f \in \mathcal{G}_{\mathcal{X}}^k$ and $g \in \mathcal{G}_{\mathcal{Y}}^k$, the H-score $\mathscr{H}(f,g)$ is maximized if and only if $f \otimes g = \zeta_{\le k}(\mathfrak{i}_{X;Y}|\mathcal{G}_{\mathcal{X}}, \mathcal{G}_{\mathcal{Y}})$.

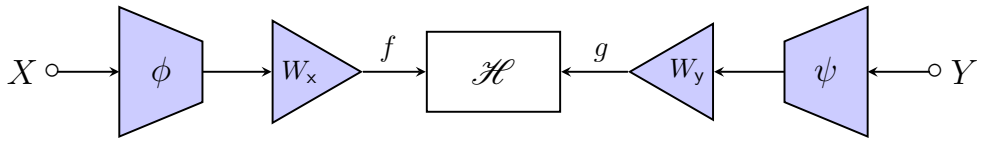


Figure 3: Features f, g as the output of linear layers. The linear layers are represented as triangle modules, with inputs ϕ, ψ, and weight matrices W_{x}, W_{y}, respectively.

As an application, we can characterize the effects of the restricted expressive power on feature learning. To begin, we consider the maximization of H-score $\mathscr{H}(f,g)$, where features f and g are k-dimensional outputs of neural networks. In particular, we assume the last layers of the networks are linear layers, which is a common network architecture design in practice. The overall network architecture is shown in Figure 3, where we express f as the composition of feature extractor $\phi \in \mathcal{F}_{\mathcal{X}}^{d_{\mathsf{x}}}$ and the last linear layer with weight matrix $W_{\mathsf{x}} \in \mathbb{R}^{k \times d_{\mathsf{x}}}$. Similarly, we represent g as the composition of $\psi \in \mathcal{F}_{\mathcal{Y}}^{d_{\mathsf{y}}}$ and the linear layer with weight $W_{\mathsf{y}} \in \mathbb{R}^{k \times d_{\mathsf{y}}}$.

Suppose we have trained the weights $W_{\mathsf{x}}, W_{\mathsf{y}}$ and the parameters in ϕ, ψ to maximize the H-score $\mathscr{H}(f,g)$, and the weights W_{x} and W_{y} have converged to their optimal values with respect to ϕ and ψ. Note that for any given ϕ, ψ, $f = W_{\mathsf{x}}\phi$ takes values from the set $\{W_{\mathsf{x}}\phi \colon W_{\mathsf{x}} \in \mathbb{R}^{k \times d_{\mathsf{x}}}\} = \operatorname{span}^k\{\phi\}$, and, similarly, $g = W_{\mathsf{y}}\psi$ takes values from $\operatorname{span}^k\{\psi\}$. Therefore, the optimal (f,g) corresponds to the solution of a constrained low-rank approximation problem, and we have $f \otimes g = \zeta_{\le k}(\mathfrak{i}_{X;Y}|\operatorname{span}\{\phi\}, \operatorname{span}\{\psi\})$.

In addition, from Proposition 5 and the orthogonality principle, we can express the approximation error as

$$\begin{aligned} \left\|\mathfrak{i}_{X;Y} - f \otimes g\right\|^2 &= \left\|\mathfrak{i}_{X;Y} - \mathfrak{i}'_{X;Y} + \mathfrak{i}'_{X;Y} - \zeta_{\le k}(\mathfrak{i}'_{X;Y})\right\|^2 \\ &= \|\mathfrak{i}_{X;Y} - \mathfrak{i}'_{X;Y}\|^2 + \|r_k(\mathfrak{i}'_{X;Y})\|^2, \end{aligned} \tag{29}$$

where $\mathfrak{i}'_{X;Y} \triangleq \Pi(\mathfrak{i}_{X;Y}; \operatorname{span}\{\phi\} \otimes \operatorname{span}\{\psi\})$. Note that (29) decomposes the overall approximation into two terms, where the first term characterizes the effects of insufficient expressive power of ϕ, ψ, and the second term characterizes the impacts of feature dimension k.

3.4 Relationship to Classification DNNs

We conclude this section by discussing a relation between the dependence approximation framework and deep neural networks, studied in Xu et al. (2022). We consider a classification task where X and Y denote the input data and the target label to predict, respectively. Then, we can interpret

the log-likelihood function of DNN as an approximation of the H-score, and thus DNN also learns strongest modes of $(X;Y)$ dependence.

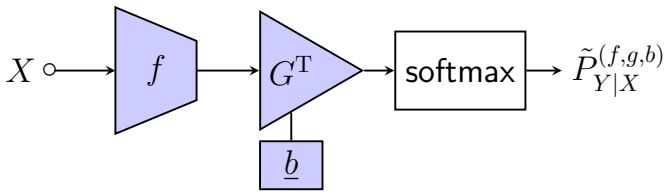


Figure 4: A classification DNN for predicting label Y based on the input X. All layers before the classification layer are represented as feature extractor f. The weight and bias associated with each class $Y = y$ are denoted by $g(y)$ and $b(y)$, respectively, which give weight matrix G^{T} and bias vector $\underline{b}$ with $G = [g(1), \ldots, g(|\mathcal{Y}|)], \underline{b} = [b(1), \ldots, g(|\mathcal{Y}|)]^{\mathrm{T}}$. The softmax module outputs a posterior probability, parameterized by f, g and b.

To begin, let $\{(x_i, y_i)\}_{i=1}^n$ denote the training data, with empirical distribution $P_{X,Y}$ as defined in (9). We depict the architecture of typical classification DNN in Figure 4, where we abstract all layers before classification layer as a k-dimensional feature extractor $f \in \mathcal{F}_{\mathcal{X}}^k$. The feature f is then processed by a classification layer with weight matrix G^{T} and the bias vector $\underline{b}$, and activated by the softmax function[9]. Without loss of generality, we assume $\mathcal{Y} = \{1, \ldots, |\mathcal{Y}|\}$, then we can represent G and $\underline{b}$ as

$$G \triangleq [g(1), \ldots, g(|\mathcal{Y}|)] \in \mathbb{R}^{k \times |\mathcal{Y}|}, \quad \underline{b} \triangleq [b(1), \ldots, b(|\mathcal{Y}|)]^{\mathrm{T}} \in \mathbb{R}^{|\mathcal{Y}|}, \tag{30}$$

where $g(y) \in \mathbb{R}^k$ and $b(y) \in \mathbb{R}$ denote the weight and bias associated with each class $Y = y$, respectively. Then, the softmax output of $(G^{\mathrm{T}} f(x) + \underline{b})$ gives a parameterized posterior

$$\tilde{P}_{Y|X}^{(f,g,b)}(y|x) \triangleq \frac{\exp(f(x) \cdot g(y) + b(y))}{\sum_{y' \in \mathcal{Y}} \exp(f(x) \cdot g(y') + b(y'))}. \tag{31}$$

The network parameters are trained to maximize[10] the resulting log-likelihood[11] function

$$\mathcal{L}(f, g, b) \triangleq \frac{1}{n} \sum_{i=1}^{n} \log \tilde{P}_{Y|X}^{(f,g,b)}(y_i|x_i) = \mathbb{E}_{(\hat{X},\hat{Y}) \sim P_{X,Y}} \left[\log \tilde{P}_{Y|X}^{(f,g,b)}(\hat{Y}|\hat{X}) \right]. \tag{32}$$

We further define $\mathcal{L}(f, g) \triangleq \max_{b \in \mathcal{F}_{\mathcal{Y}}} \mathcal{L}(f, g, b)$, by setting the bias b to its optimal value with respect to given f and g. It can be verified that $\mathcal{L}(f, g)$ depends only on the centered versions of f and g, formalized as follows. A proof is provided in Appendix C.5.

Property 3 *For all $k \geq 1$ and $f \in \mathcal{F}_{\mathcal{X}}^k$, $g \in \mathcal{F}_{\mathcal{Y}}^k$, we have $\mathcal{L}(f, g) = \mathcal{L}(\tilde{f}, \tilde{g})$, where we have defined $\tilde{f} \in \mathcal{F}_{\mathcal{X}}^k, \tilde{g} \in \mathcal{F}_{\mathcal{Y}}^k$ as $\tilde{f} \triangleq \Pi\left(f; \tilde{\mathcal{F}}_{\mathcal{X}}\right), \tilde{g} \triangleq \Pi\left(g; \tilde{\mathcal{F}}_{\mathcal{Y}}\right)$, i.e., $\tilde{f}(x) = f(x) - \mathbb{E}[f(X)]$, $\tilde{g}(y) = g(y) - \mathbb{E}[g(Y)]$, for all $x \in \mathcal{X}, y \in \mathcal{Y}$.*

9. The softmax function is defined such that, for all $k > 1$ and each $\underline{v} = (v_1, \ldots, v_k)^{\mathrm{T}} \in \mathbb{R}^k$, we have $\mathrm{softmax}(\underline{v}) \in \mathbb{R}^k$, with each i-th entry being $[\mathrm{softmax}(\underline{v})]_i \triangleq \frac{\exp(v_i)}{\sum_{j=1}^k \exp(v_j)}$, $i \in [k]$.

10. This is equivalent to minimizing $-\mathcal{L}(f, g, b)$, i.e., the log loss (cross entropy loss).

11. Throughout our development, all logarithms are base e, i.e., natural.

Therefore, it is without loss of generalities to restrict our discussions to zero-mean f and g. Specifically, we can verify that for the trivial choice of feature $f = 0$, the resulting likelihood function is $\mathcal{L}(0, g) = \mathcal{L}(0, 0) = -H(Y)$, achieved when the posterior distribution satisfies $\tilde{P}_{Y|X}^{(0,g,b)} = P_Y$, where $H(\cdot)$ denotes the Shannon entropy. In general, we have the following characterization of $\mathcal{L}(f, g)$, which extends (Xu et al., 2022, Theorem 4). A proof is provided in Appendix C.6.

Proposition 14 *Suppose X and Y are ϵ-dependent. For all $k \geq 1$, and $f \in \tilde{\mathcal{F}}_{\mathcal{X}}^k$, $g \in \tilde{\mathcal{F}}_{\mathcal{Y}}^k$, if $\mathcal{L}(f, g) \geq \mathcal{L}(0, 0) = -H(Y)$, then we have $\|f \otimes g\| = O(\epsilon)$, and*

$$\mathcal{L}(f, g) = \mathcal{L}(0, 0) + \underbrace{\frac{1}{2} \cdot \left(\|\mathfrak{i}_{X;Y}\|^2 - \left\|\mathfrak{i}_{X;Y} - f \otimes g\right\|^2 \right)}_{=\mathscr{H}(f,g)} + o(\epsilon^2) \tag{33}$$

which is maximized if and only if $f \otimes g = \zeta_{\leq k}(\mathfrak{i}_{X,Y}) + o(\epsilon)$.

From Proposition 14, the H-score $\mathscr{H}(f, g)$ coincides with likelihood function $\mathcal{L}(f, g)$ in the local regime. For a fully expressive feature extractor f of dimension k, the optimal feature f and weight matrix G^{T} are approximating the rank-k approximation of $(X; Y)$ dependence. In this sense, the weight matrix G^{T} in classification DNN essentially characterizes a feature of the label Y, with a role symmetric to feature extractor f. However, unlike the H-score implementation, the classification DNN is restricted to categorical Y to make the softmax function (31) computable.

Remark 15 *For general X, Y, the optimal features f, g that optimize the likelihood function* (32) *can deviate from the low-rank approximation characterization in Proposition 14 and have different behaviors. See Xu et al. (2018) for detailed discussions.*

4. Nesting Technique for Dependence Decomposition

In multivariate learning applications, it is often difficult to summarize the statistical dependence as some bivariate dependence. Instead, the statistical dependence of interest is typically only a component decomposed from the original dependence. In this section, we introduce a nesting technique, which allows us to implement such dependence decomposition operations by training corresponding neural feature extractors. For the ease of presentation, we adopt the bivariate setting introduced previously and consider the feature geometry on $\mathcal{F}_{\mathcal{X}\times\mathcal{Y}}$ with metric distribution $P_X P_Y$. We will discuss the multivariate extensions in later sections.

4.1 Nesting Configuration and Nested H-score

The nesting technique is a systematic approach to learn features representing projected dependence components or their modal decomposition. In particular, for a given dependence component of interest, we can construct corresponding training objective for learning the dependence component. The resulting training objective is an aggregation of different H-scores, where the inputs to these H-scores are features forming a nested structure. We refer to such functions as the *nested H-scores*. To specify a nested H-score, we introduce its configuration, referred to as the *nesting configuration*, defined as follows.

Definition 16 *Given $\mathcal{X}, \mathcal{Y}$ and $k \geq l \geq 1$, we define an l-level nesting configuration for k-dimensional features as the tuple $\left\{(d_1, \ldots, d_l);\ (\mathcal{G}_{\mathcal{X}}^{(1)}, \ldots, \mathcal{G}_{\mathcal{X}}^{(l)});\ \mathcal{G}_{\mathcal{Y}}\right\}$, where*

- *$(d_1, \cdots, d_l)$ is a sequence with $d_i > 0$ and $\sum_{i=1}^{l} d_i = k$;*

- $\left(\mathcal{G}_{\mathcal{X}}^{(1)}, \ldots, \mathcal{G}_{\mathcal{X}}^{(l)}\right)$ *is an increasing sequence of* l *subspaces of* $\mathcal{F}_{\mathcal{X}}$*:* $\mathcal{G}_{\mathcal{X}}^{(1)} \subset \cdots \subset \mathcal{G}_{\mathcal{X}}^{(l)}$*;*
- $\mathcal{G}_{\mathcal{Y}}$ *is a subspace of* $\mathcal{F}_{\mathcal{Y}}$*.*

Nested H-score Given a nesting configuration $\mathcal{C} = \left\{(d_1, \ldots, d_l);\ \left(\mathcal{G}_{\mathcal{X}}^{(1)}, \ldots, \mathcal{G}_{\mathcal{X}}^{(l)}\right);\ \mathcal{G}_{\mathcal{Y}}\right\}$ for k-dimensional features, the associated nested H-score is a function of k-dimensional feature pair f and g, which we denote by $\mathscr{H}(f, g; \mathcal{C})$, specified as follows. To begin, let us define $k_i \triangleq \sum_{j=1}^{i} d_j$ for each $0 \leq i \leq l$, representing the total dimension up to i-th level. Then, we define the domain of $\mathscr{H}(f, g; \mathcal{C})$, denoted by $\mathrm{dom}(\mathcal{C})$, as

$$\mathrm{dom}(\mathcal{C}) \triangleq \left\{(f, g)\colon f \in \mathcal{F}_{\mathcal{X}}^{k}, g \in \mathcal{G}_{\mathcal{Y}}^{k}, f_j \in \mathcal{G}_{\mathcal{X}}^{(i)},\ \text{for all } k_{i-1} < j \leq k_i\right\}. \tag{34}$$

Then, for $(f, g) \in \mathrm{dom}(\mathcal{C})$ and each $i \in [l]$, we obtain the H-score $\mathscr{H}(f_{[k_i]}, g_{[k_i]})$ by taking the first k_i dimensions of f, g. We define the nested H-score $\mathscr{H}(f, g; \mathcal{C})$ by taking the sum of these l H-scores,

$$\mathscr{H}\left(f, g; \mathcal{C}\right) \triangleq \sum_{i=1}^{l} \mathscr{H}(f_{[k_i]}, g_{[k_i]}), \qquad (f, g) \in \mathrm{dom}(\mathcal{C}). \tag{35}$$

From (35), the nested H-score aggregates different H-scores with nested input features. The nested structure of features is specified by the increasing sequence of dimension indices: $[k_1] \subset \cdots \subset [k_l] = [k]$, determined by the sequence $(d_1, \ldots, d_l)$. The domain of features is specified by subspaces in the configuration. When $\mathcal{G}_{\mathcal{X}}^{(i)} = \mathcal{G}_{\mathcal{X}}$ for all $i \in [l]$, we can simply write the configuration as $\mathcal{C} = \{(d_1, \ldots, d_l);\ \mathcal{G}_{\mathcal{X}};\ \mathcal{G}_{\mathcal{Y}}\}$ without ambiguity. In particular, we can represent the original H-score for k-dimensional input features as a nested H-score configured by $\{k;\ \mathcal{F}_{\mathcal{X}};\ \mathcal{F}_{\mathcal{Y}}\}$.

Remark 17 *Note that the nested H-score* (35) *is obtained by using a sum function to aggregate different H-score terms* $\mathscr{H}(f_{[k_i]}, g_{[k_i]})$*,* $i = 1, \ldots, l$*. We shall comment that the choice of such aggregation functions is not unique. Generally, for an* l*-level nesting configuration, we can apply any differentiable* $\Gamma\colon \mathbb{R}^l \to \mathbb{R}$ *as an aggregation function if* Γ *is strictly increasing in each argument. The aggregated result* $\Gamma(\mathscr{H}(f_{[k_1]}, g_{[k_1]}), \ldots, \mathscr{H}(f_{[k_l]}, g_{[k_l]}))$ *defines a nested H-score that satisfies the same collection of properties. For the ease of presentation, we adopt the sum form* (35) *throughout our development, but also provide general discussions in Appendix B for completeness.*

Remark 18 *By symmetry, we can also define the configuration* $\left\{(d_1, \ldots, d_l);\ \mathcal{G}_{\mathcal{X}};\ \left(\mathcal{G}_{\mathcal{Y}}^{(1)}, \ldots, \mathcal{G}_{\mathcal{Y}}^{(l)}\right)\right\}$ *and the associated nested H-score, for subspaces* $\mathcal{G}_{\mathcal{X}}$ *of* $\mathcal{F}_{\mathcal{X}}$ *and* $\mathcal{G}_{\mathcal{Y}}^{(1)} \subset \cdots \subset \mathcal{G}_{\mathcal{Y}}^{(l)}$ *of* $\mathcal{F}_{\mathcal{Y}}$*.*

Refinements of Nesting Configuration Given a nesting configuration for k-dimensional features $\mathcal{C} = \left\{(d_1, \ldots, d_l);\ \left(\mathcal{G}_{\mathcal{X}}^{(1)}, \ldots, \mathcal{G}_{\mathcal{X}}^{(l)}\right);\ \mathcal{G}_{\mathcal{Y}}\right\}$, the sequence $(d_1, \ldots, d_l)$ defines a partition that separates the k dimensions into l different groups. By refining such partition, we can construct new configurations with higher levels, which we refer to as refined configurations. In particular, the finest refinement corresponds to the partition where each group has only one dimension. We use $\mathcal{C}^\star$ to denote the finest refinement of $\mathcal{C}$, given by

$$\mathcal{C}^\star \triangleq \left\{(1)^k;\ \left(\left(\mathcal{G}_{\mathcal{X}}^{(1)}\right)^{d_1}, \ldots, \left(\mathcal{G}_{\mathcal{X}}^{(l)}\right)^{d_l}\right);\ \mathcal{G}_{\mathcal{Y}}\right\}, \tag{36}$$

where we have used $(1)^k$ to denote the all-one sequence of length k, and where $\left(\left(\mathcal{G}_{\mathcal{X}}^{(1)}\right)^{d_1}, \ldots, \left(\mathcal{G}_{\mathcal{X}}^{(l)}\right)^{d_l}\right)$ represents the length-k sequence starting with d_1 terms of $\mathcal{G}_{\mathcal{X}}^{(1)}$, followed by d_2 terms of $\mathcal{G}_{\mathcal{X}}^{(2)}$,

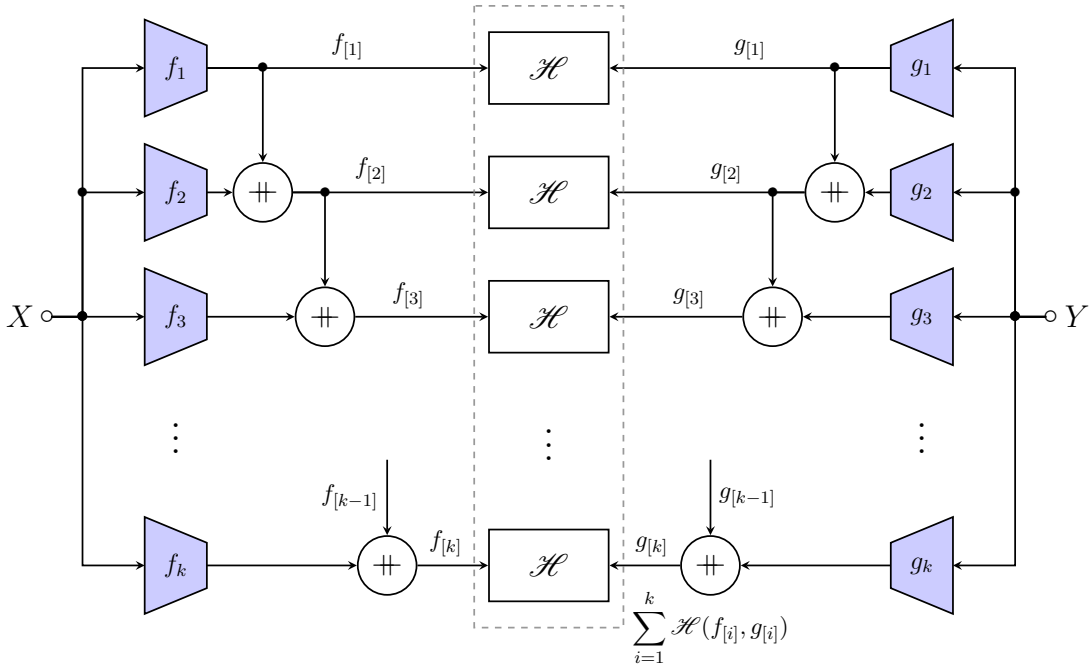


Figure 5: Nesting technique for modal decomposition: the nested H-score is computed with a nested architecture, where "$+\!\!+$" denotes the concatenation operation of two features.

up to d_l terms of $\mathcal{G}_{\mathcal{X}}^{(l)}$. From (34), such refinements do not change the domain, and we have $\mathrm{dom}(\mathcal{C}^\star) = \mathrm{dom}(\mathcal{C})$. The corresponding nested H-score is

$$\mathscr{H}\left(f, g; \mathcal{C}^\star\right) = \sum_{i=1}^{k} \mathscr{H}(f_{[i]}, g_{[i]}), \quad (f, g) \in \mathrm{dom}(\mathcal{C}). \tag{37}$$

4.2 Nesting Technique for Modal Decomposition

We then demonstrate the application of nesting technique in learning modal decomposition. Given k-dimensional features f, g, we consider the nesting configuration $\{(1)^k; \mathcal{F}_{\mathcal{X}}; \mathcal{F}_{\mathcal{Y}}\}$, which can also be obtained from the original H-score by the refinement (36): $\{(1)^k; \mathcal{F}_{\mathcal{X}}; \mathcal{F}_{\mathcal{Y}}\} = \{k; \mathcal{F}_{\mathcal{X}}; \mathcal{F}_{\mathcal{Y}}\}^\star$. The corresponding nested H-score is the sum of k H-scores:

$$\mathscr{H}(f, g; \{(1)^k; \mathcal{F}_{\mathcal{X}}; \mathcal{F}_{\mathcal{Y}}\}) = \sum_{i=1}^{k} \mathscr{H}(f_{[i]}, g_{[i]}). \tag{38}$$

Note that from Property 2, for each $i \in [k]$, the H-score $\mathscr{H}(f_{[i]}, g_{[i]})$ is maximized if and only if $f_{[i]} \otimes g_{[i]} = \zeta_{\le i}(\mathfrak{i}_{X;Y})$. Therefore, all k terms of H-scores are maximized simultaneously, if and only if we have $f_{[i]} \otimes g_{[i]} = \zeta_{\le i}(\mathfrak{i}_{X;Y})$ for all $i \in [k]$. By definition, this is also equivalent to

$$f_i \otimes g_i = \zeta_i(\mathfrak{i}_{X;Y}), \quad i \in [k]. \tag{39}$$

Hence, the nested H-score $\mathscr{H}(f, g; \{(1)^k; \mathcal{F}_{\mathcal{X}}; \mathcal{F}_{\mathcal{Y}}\})$ is maximized if and only if we have (39), which gives the top k modes of $(X; Y)$ dependence. In practice, we can compute the nested H-score by using a nested architecture as shown in Figure 5, where we have used the "$+\!\!+$" symbol to indicate

the concatenation of two vectors, i.e., $\underline{u} + \underline{v} \triangleq [\frac{\underline{u}}{\underline{v}}]$ for two column vectors $\underline{u}, \underline{v}$. By maximizing the nested H-score, we can use (39) to retrieve each i-th dependence mode from corresponding feature pair (f_i, g_i), for $i \in [k]$.

Compared with the features learned in Section 3, the nesting technique provides several new applications. First, from Fact 3, the learned features f and g have orthogonal dimensions, i.e., different dimensions are uncorrelated. In addition, from (39), we can compute the energy contained in each i-th dependence mode, via $\| \zeta_i(\mathfrak{i}_{X;Y})\|^2 = \|f_i \otimes g_i\|^2 = \mathbb{E}\left[f_i^2(X)\right] \cdot \mathbb{E}\left[g_i^2(Y)\right]$, for $i \in [k]$. This provides a spectrum of $(X;Y)$ dependence and characterizes the usefulness or contribution of each dimension. Similarly, we can retrieve top k maximal correlation functions f_i^*, g_i^* and coefficients σ_i, by using the relations [cf. (19) and Corollary 6]

$$f_i^* = \frac{f_i}{\sqrt{\mathbb{E}\left[f_i^2(X)\right]}}, \quad g_i^* = \frac{g_i}{\sqrt{\mathbb{E}\left[g_i^2(Y)\right]}}, \quad \sigma_i = \sqrt{\mathbb{E}\left[f_i^2(X)\right] \cdot \mathbb{E}\left[g_i^2(Y)\right]}, \qquad i \in [k]. \tag{40a}$$

Nested Optimality From (39), for any $d \le k$, we can represent the optimal rank-d approximation of $\mathfrak{i}_{X;Y}$ as $\zeta_{\le d}(\mathfrak{i}_{X;Y}) = f_{[d]} \otimes g_{[d]}$, which corresponds to top d-dimensions of learned features. We refer to this property as nested optimality: the learned features give a collection of optimal solutions for different dimensions, with a nested structure. This nested optimality provides a convenient and principled feature selection *on the fly*: we can obtain the optimal selection of d feature pairs by simply taking the top d dimensions of learned features. It is worth noting that, this optimal *selection* indeed gives the optimal d-dimensional feature *extraction*, corresponding to the most significant d modes of the $(X;Y)$ dependence. This equivalence is hardly guaranteed in typical feature selection approaches. In practice, we can choose the dimension d based on the dependence spectrum, such that selected features capture sufficient amount of dependence information, and then take $f_{[d]}, g_{[d]}$ for further processing.

We can readily extend the discussion to constrained modal decomposition problems. Let $\mathcal{G}_\mathcal{X}$ and $\mathcal{G}_\mathcal{Y}$ be subspaces of $\mathcal{F}_\mathcal{X}$ and $\mathcal{F}_\mathcal{Y}$, respectively. Then, the nested H-score $\mathscr{H}(f, g; \{(1)^k; \mathcal{G}_\mathcal{X}; \mathcal{G}_\mathcal{Y}\})$ defined for k-dimensional features $f \in \mathcal{G}_\mathcal{X}^k$, $g \in \mathcal{G}_\mathcal{Y}^k$, is maximized if and only if

$$f_i \otimes g_i = \zeta_i(\mathfrak{i}_{X;Y} | \mathcal{G}_\mathcal{X}, \mathcal{G}_\mathcal{Y}), \quad \text{for all } i \in [k]. \tag{41}$$

From (41), we can establish a similar nested optimality in the constrained case. In particular, when $\mathcal{G}_\mathcal{X}$ and $\mathcal{G}_\mathcal{Y}$ correspond to the collection of features that can be expressed by neural feature extractors, the result also characterizes the effects of restricted expressive power of neural networks (cf. Section 3.3). Specifically, from (41), when we use feature extractors with restricted expressive power, we can still guarantee the learned features have uncorrelated dimensions.

4.3 Nesting Technique for Projection

With the nesting technique, we can also operate projections of statistical dependence in feature spaces. Such operations are the basis of multivariate dependence decomposition, which we will detail in the following sections.

To begin, let $\mathcal{G}_\mathcal{X}$ denote a subspace of $\mathcal{F}_\mathcal{X}$. Then, from $\mathcal{F}_\mathcal{X} = \mathcal{G}_\mathcal{X} \boxplus (\mathcal{F}_\mathcal{X} \boxminus \mathcal{G}_\mathcal{X})$, we obtain an orthogonal decomposition of function space

$$\mathcal{F}_{\mathcal{X}\times\mathcal{Y}} = \mathcal{F}_\mathcal{X} \otimes \mathcal{F}_\mathcal{Y} = (\mathcal{G}_\mathcal{X} \otimes \mathcal{F}_\mathcal{Y}) \boxplus ((\mathcal{F}_\mathcal{X} \boxminus \mathcal{G}_\mathcal{X}) \otimes \mathcal{F}_\mathcal{Y}). \tag{42}$$

Therefore, by projecting the statistical dependence $\mathfrak{i}_{X;Y}$ to these function spaces, we obtain its orthogonal decomposition [cf. Fact 2]

$$\mathfrak{i}_{X;Y} = \Pi\left(\mathfrak{i}_{X;Y}; \mathcal{G}_\mathcal{X} \otimes \mathcal{F}_\mathcal{Y}\right) + \Pi\left(\mathfrak{i}_{X;Y}; (\mathcal{F}_\mathcal{X} \boxminus \mathcal{G}_\mathcal{X}) \otimes \mathcal{F}_\mathcal{Y}\right). \tag{43}$$

In particular, the first term $\Pi\left(\mathfrak{i}_{X;Y}; \mathcal{G}_X \otimes \mathcal{F}_Y\right)$ characterizes the dependence component aligned with the subspace $\mathcal{G}_X$, and the second term represents the component orthogonal to $\mathcal{G}_X$. For convenience, we denote these two dependence components by $\pi(\mathfrak{i}_{X;Y})$ and $\pi_\perp(\mathfrak{i}_{X;Y})$, respectively, and demonstrate the geometry of the decomposition in Figure 6.

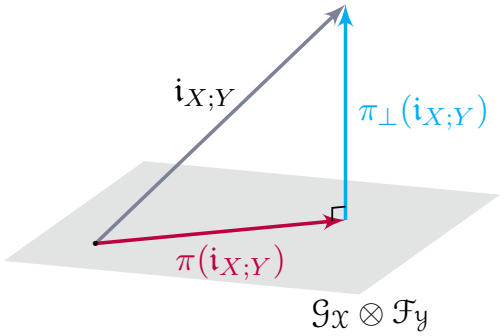


Figure 6: Orthogonal decomposition of the CDK function $\mathfrak{i}_{X;Y}$: $\pi(\mathfrak{i}_{X;Y})$ denotes the projection onto the plane $\mathcal{G}_X \otimes \mathcal{F}_Y$, and $\pi_\perp(\mathfrak{i}_{X;Y})$ denotes the residual, orthogonal to the plane.

In general, the information carried by decomposed dependence components depends on the choices of subspace $\mathcal{G}_X$, which varies in different learning settings. In spite of such differences, we can learn the decomposition with a unified procedure, which we demonstrate as follows.

To begin, we consider the feature representations of the dependence components. For example, by applying the rank-k approximation on the orthogonal component $\pi_\perp(\mathfrak{i}_{X;Y})$, we obtain

$$\zeta_{\le k}(\pi_\perp(\mathfrak{i}_{X;Y})) = \zeta_{\le k}(\Pi\left(\mathfrak{i}_{X;Y}; (\mathcal{F}_X \boxminus \mathcal{G}_X) \otimes \mathcal{F}_Y\right)) = \zeta_{\le k}(\mathfrak{i}_{X;Y} | \mathcal{F}_X \boxminus \mathcal{G}_X, \mathcal{F}_Y),$$

which can be represented as a pair of k-dimensional features. To learn such feature representations, we introduce the two-level nesting configuration

$$\mathcal{C}_\pi \triangleq \{(\bar{k}, k);\ (\mathcal{G}_X, \mathcal{F}_X);\ \mathcal{F}_Y\} \tag{44}$$

for some feature dimensions $\bar{k}, k \ge 1$. The corresponding nested H-score is

$$\mathscr{H}\left(\begin{bmatrix}\bar{f}\\ f\end{bmatrix}, \begin{bmatrix}\bar{g}\\ g\end{bmatrix}; \mathcal{C}_\pi\right) = \mathscr{H}(\bar{f}, \bar{g}) + \mathscr{H}\left(\begin{bmatrix}\bar{f}\\ f\end{bmatrix}, \begin{bmatrix}\bar{g}\\ g\end{bmatrix}\right), \tag{45}$$

defined on the domain [cf. (34)]

$$\mathrm{dom}(\mathcal{C}_\pi) = \left\{\left(\begin{bmatrix}\bar{f}\\ f\end{bmatrix}, \begin{bmatrix}\bar{g}\\ g\end{bmatrix}\right) : \bar{f} \in \mathcal{G}_X^{\bar{k}}, \bar{g} \in \mathcal{F}_Y^{\bar{k}}, f \in \mathcal{F}_X^{k}, g \in \mathcal{F}_Y^{k}\right\}, \tag{46}$$

where for convenience, we explicitly express the first-level features as $\bar{f}, \bar{g}$, both of dimension $\bar{k}$. We can use a nested network architecture to compute the nested H-score (45), as shown in Figure 7.

To see the roles of the two H-score terms in (45), note that if we maximize only the first term $\mathscr{H}(\bar{f}, \bar{g})$ of the nested H-score over the domain (46), we will obtain the solution to a constrained dependence approximation problem (cf. Section 3.3): $\bar{f} \otimes \bar{g} = \zeta_{\le \bar{k}}(\mathfrak{i}_{X;Y} | \mathcal{G}_X, \mathcal{F}_Y)$. Specifically, if $\bar{k}$ is sufficiently large, we would get $\bar{f} \otimes \bar{g} = \Pi\left(\mathfrak{i}_{X;Y}; \mathcal{G}_X \otimes \mathcal{F}_Y\right) = \pi(\mathfrak{i}_{X;Y})$, which gives the aligned component. With such $\bar{f}$ and $\bar{g}$, we can express the second H-score term as

$$\mathscr{H}\left(\begin{bmatrix}\bar{f}\\ f\end{bmatrix}, \begin{bmatrix}\bar{g}\\ g\end{bmatrix}\right) = \frac{1}{2} \cdot \Big[\|\mathfrak{i}_{X;Y}\|^2 - \|\underbrace{\mathfrak{i}_{X;Y} - \bar{f} \otimes \bar{g}}_{=\pi_\perp(\mathfrak{i}_{X;Y})} - f \otimes g\|^2\Big].$$

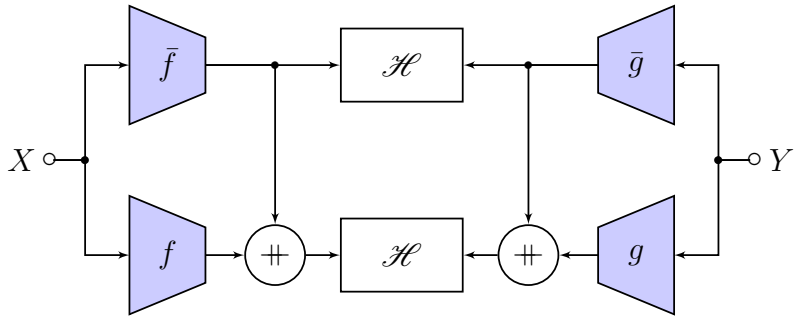


Figure 7: Two-level Nested Network With Features $\begin{bmatrix}\bar{f}\\ f\end{bmatrix}$ and $\begin{bmatrix}\bar{g}\\ g\end{bmatrix}$

Therefore, if we maximize the second H-score over only f and g, we would get the orthogonal dependence component: $f \otimes g = \zeta_{\leq k}(\pi_{\perp}(\mathfrak{i}_{X;Y}))$. This gives a two-phase training strategy for computing the decomposition (43).

In contrast, the nested H-score (45) provides a single training objective to obtain both dependence components simultaneously. We formalize the result as the following theorem, of which a proof is provided in Appendix C.7.

Theorem 19 *Given* $\bar{k} \geq \operatorname{rank}(\Pi(\mathfrak{i}_{X;Y}; \mathcal{G}_{\mathcal{X}} \otimes \mathcal{F}_{\mathcal{Y}}))$, $\mathscr{H}\left(\begin{bmatrix}\bar{f}\\ f\end{bmatrix}, \begin{bmatrix}\bar{g}\\ g\end{bmatrix}; \mathcal{C}_{\pi}\right)$ *is maximized if and only if*

$$\bar{f} \otimes \bar{g} = \Pi(\mathfrak{i}_{X;Y}; \mathcal{G}_{\mathcal{X}} \otimes \mathcal{F}_{\mathcal{Y}}), \tag{47a}$$
$$f \otimes g = \zeta_{\leq k}(\mathfrak{i}_{X;Y} | \mathcal{F}_{\mathcal{X}} \boxminus \mathcal{G}_{\mathcal{X}}, \mathcal{F}_{\mathcal{Y}}). \tag{47b}$$

We can further consider the modal decomposition of dependence components, to obtain features with nested optimality. To learn such features, it suffices to consider the refined configuration $\mathcal{C}_{\pi}^{\star} = \left\{(1)^{\bar{k}+k};\ (\mathcal{G}_{\mathcal{X}}^{\bar{k}}, \mathcal{F}_{\mathcal{X}}^{k});\ \mathcal{F}_{\mathcal{Y}}\right\}$, which we formalize as follows. A proof is provided in Appendix C.8.

Theorem 20 *Given* $\bar{k} \geq \operatorname{rank}(\Pi(\mathfrak{i}_{X;Y}; \mathcal{G}_{\mathcal{X}} \otimes \mathcal{F}_{\mathcal{Y}}))$, $\mathscr{H}\left(\begin{bmatrix}\bar{f}\\ f\end{bmatrix}, \begin{bmatrix}\bar{g}\\ g\end{bmatrix}; \mathcal{C}_{\pi}^{\star}\right)$ *is maximized if and only if*

$$\bar{f}_i \otimes \bar{g}_i = \zeta_i(\mathfrak{i}_{X;Y} | \mathcal{G}_{\mathcal{X}}, \mathcal{F}_{\mathcal{Y}}) \quad \textit{for all } i \in [\bar{k}], \tag{48a}$$
$$f_i \otimes g_i = \zeta_i(\mathfrak{i}_{X;Y} | \mathcal{F}_{\mathcal{X}} \boxminus \mathcal{G}_{\mathcal{X}}, \mathcal{F}_{\mathcal{Y}}) \quad \textit{for all } i \in [k]. \tag{48b}$$

4.4 Learning With Orthogonality Constraints

We conclude this section by discussing an application of the nesting technique, where the goal is to learn optimal features uncorrelated to some given features. Such settings arise in many learning scenarios, where the given features correspond to some prior knowledge. For instance, when the given features are extracted from pre-trained models, we can use this formulation to avoid information overlapping and extract only the part not carried by these models. Another example is to extract privacy-preserving features, where we use the given features to indicate the sensitive information.

In feature geometry, the uncorrelatedness conditions correspond to orthogonality constraints, and we can formalize the learning problem as follows. Given a $\bar{k}$-dimensional feature $\phi \in \mathcal{F}_{\mathcal{X}}^{\bar{k}}$, our goal is to learn k-dimensional feature f from X for inferring Y, with the constraint that

$\text{span}\{f\} \perp \text{span}\{\phi\}$, i.e., $\mathbb{E}[f_i(X)\phi_j(X)] = \langle f_i, \phi_j \rangle = 0$, for all $i \in [k], j \in [\bar{k}]$. We therefore consider the constrained low-rank approximation problem

$$\underset{f \in \mathcal{F}_{\mathcal{X}}^k, g \in \mathcal{F}_{\mathcal{Y}}^k :\ \text{span}\{f\} \perp \text{span}\{\phi\}}{\text{minimize}} \quad \left\| \mathfrak{i}_{X;Y} - f \otimes g \right\|. \tag{49}$$

We can demonstrate that the solution to (49) corresponds to learning the decomposition (43), with the choice $\mathcal{G}_{\mathcal{X}} = \text{span}\{\phi\}$. To see this, we rewrite (49) as

$$\mathfrak{i}_{X;Y} = \pi_\phi(\mathfrak{i}_{X;Y}) + (\mathfrak{i}_{X;Y} - \pi_\phi(\mathfrak{i}_{X;Y}))\,. \tag{50}$$

where we have denoted the aligned component $\pi_\phi(\mathfrak{i}_{X;Y}) \triangleq \Pi\left(\mathfrak{i}_{X;Y}; \text{span}\{\phi\} \otimes \mathcal{F}_{\mathcal{Y}}\right)$. In addition, note that the orthogonality constraint of (49) is $f \in (\mathcal{F}_{\mathcal{X}} \boxminus \text{span}\{\phi\})^k, g \in \mathcal{F}_{\mathcal{Y}}^k$. Therefore, it follows from Proposition 5 that the solution to (49) is

$$\begin{aligned} f \otimes g = \zeta_{\le k}(\mathfrak{i}_{X;Y} | \mathcal{F}_{\mathcal{X}} \boxminus \text{span}\{\phi\}, \mathcal{F}_{\mathcal{Y}}) &= \zeta_{\le k}(\Pi\left(\mathfrak{i}_{X;Y}; (\mathcal{F}_{\mathcal{X}} \boxminus \text{span}\{\phi\}) \otimes \mathcal{F}_{\mathcal{Y}}\right)) \\ &= \zeta_{\le k}\left(\mathfrak{i}_{X;Y} - \pi_\phi(\mathfrak{i}_{X;Y})\right), \end{aligned} \tag{51}$$

where to obtain the last equality we have used the orthogonal decomposition (50), as well as the fact that $(\mathcal{F}_{\mathcal{X}} \boxminus \text{span}\{\phi\}) \otimes \mathcal{F}_{\mathcal{Y}} = \mathcal{F}_{\mathcal{X}\times\mathcal{Y}} \boxminus (\text{span}\{\phi\} \otimes \mathcal{F}_{\mathcal{Y}})$.

Remark 21 *From the decomposition* (50)*, we can characterize the amount of dependence information captured by feature ϕ, as the energy $\|\pi_\phi(\mathfrak{i}_{X;Y})\|^2$. This quantity (with a $1/2$ scaling factor) is also referred to as the* single-sided H-score *(Xu and Huang, 2020; Xu et al., 2022) of ϕ, due to the connection:* $\max_{\bar{g} \in \mathcal{F}_{\mathcal{Y}}^{\bar{k}}} \mathscr{H}(\phi, \bar{g}) = \frac{1}{2}\|\pi_\phi(\mathfrak{i}_{X;Y})\|^2$.

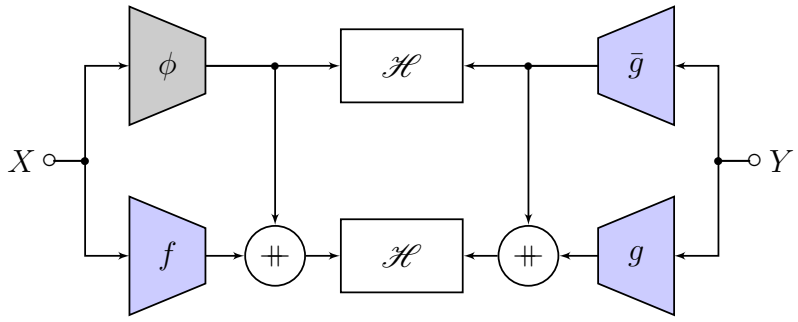


Figure 8: Nesting technique for learning features orthogonal to feature ϕ, where the ϕ block is frozen during learning. The feature ϕ can be given either in its analytical expressions, or as a pretrained neural network.

To learn the features (51), we can apply the nesting technique and maximize the nested H-score configured by $\mathcal{C}_\pi$ with $\mathcal{G}_{\mathcal{X}} = \text{span}\{\phi\}$. Specifically, from Theorem 19, $\bar{f} = \phi$ is already in the optimal solution set. Therefore, we can fix $\bar{f}$ to $\bar{f} = \phi$, and optimize

$$\mathscr{H}\left(\begin{bmatrix}\bar{f}\\ f\end{bmatrix}, \begin{bmatrix}\bar{g}\\ g\end{bmatrix}; \mathcal{C}_\pi\right)\Big|_{\bar{f}=\phi} = \mathscr{H}(\phi, \bar{g}) + \mathscr{H}\left(\begin{bmatrix}\phi\\ f\end{bmatrix}, \begin{bmatrix}\bar{g}\\ g\end{bmatrix}\right) \tag{52}$$

over $\bar{g} \in \mathcal{F}_{\mathcal{Y}}^{\bar{k}}, f \in \mathcal{F}_{\mathcal{X}}^{k}$, and $g \in \mathcal{F}_{\mathcal{Y}}^{k}$. We can compute the objective (52) by the nested network structure as shown in Figure 8.

It is also worth noting that from Proposition 14, we can also interpret the solution to (49) as the features extracted by classification DNNs subject to the same orthogonality constraints. However, compared with the H-score optimization, putting such equality constraints in DNN training typically requires non-trivial implementation.

5. Learning With Side Information

In this section, we study a multivariate learning problem involving external knowledge and demonstrate learning algorithm design based on the nesting technique. Specifically, we consider the problem of learning features from X to infer Y, and assume some external knowledge S is available for the inference. We refer to S as the side information, which corresponds to extra data sources for facilitating the inference. In particular, we consider the setting where we cannot obtain (X, S) joint pair during the information processing, e.g., X and S are collected and processed by different agents in a distributed system. Otherwise, we can apply the bivariate dependence learning framework, by treating the (X, S) pair as a new variable and directly learn features to predict Y.

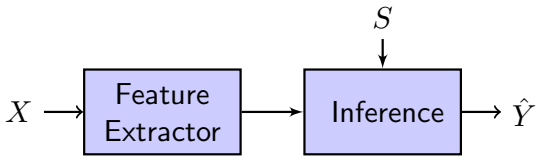


Figure 9: Learning Setting With Side Information S

We depict this learning setting in Figure 9, where the inference is based on the both features extracted from X and the side information S. Our goal is to design an efficient feature extractor which carries only the information not captured by S. In addition, we need to design a fusion mechanism for the inference module to combine such features with the side information S, and provide inference results conditioned on the side information.

Let $P_{X,S,Y}$ denote the joint distribution of X, S, Y. Throughout our development in this section, we consider the feature geometry on $\mathcal{F}_{\mathcal{X}\times\mathcal{S}\times\mathcal{Y}}$ with the metric distribution $R_{X,S,Y} \triangleq P_X P_{S,Y}$.

5.1 Dependence Decomposition and Feature Learning

To begin, we represent the joint dependence in function space as the CDK function $\mathfrak{i}_{X;S,Y} \in \mathcal{F}_{\mathcal{X}\times\mathcal{S}\times\mathcal{Y}}$. Since the side information S is provided for the inference, we focus on the dependence between X and target Y not captured by the side information. To this end, we separate the $(X;S)$ dependence from the joint dependence, by considering the orthogonal decomposition of function space [cf. Fact 1 and (3)]:

$$\mathcal{F}_{\mathcal{X}\times\mathcal{S}\times\mathcal{Y}} = \mathcal{F}_{\mathcal{X}\times\mathcal{S}} \boxplus \mathcal{F}_{\mathcal{Y}|(\mathcal{X}\times\mathcal{S})}. \tag{53}$$

This induces an orthogonal decomposition of the joint dependence

$$\mathfrak{i}_{X;S,Y} = \pi_{\mathsf{M}}(\mathfrak{i}_{X;S,Y}) + \pi_{\mathsf{C}}(\mathfrak{i}_{X;S,Y}), \tag{54}$$

where we have defined $\pi_{\mathsf{M}}(\gamma) \triangleq \Pi(\gamma; \mathcal{F}_{\mathcal{X}\times\mathcal{S}})$ and $\pi_{\mathsf{C}}(\gamma) \triangleq \Pi\left(\gamma; \mathcal{F}_{\mathcal{Y}|(\mathcal{X}\times\mathcal{S})}\right)$ for all $\gamma \in \mathcal{F}_{\mathcal{X}\times\mathcal{S}\times\mathcal{Y}}$. We characterize the decomposed components as follows, a proof of which is provided in Appendix C.9.

Proposition 22 *We have* $\pi_{\mathsf{M}}(\mathfrak{i}_{X;S,Y}) = \mathfrak{i}_{X;S} = \tilde{\ell}_{P^{\mathsf{M}}_{X,S,Y}}$, *where* $P^{\mathsf{M}}_{X,S,Y} \triangleq P_{X|S}P_SP_{Y|S}$.

From Proposition 22, we have $\pi_{\mathsf{M}}(\mathfrak{i}_{X;S,Y}) = \mathfrak{i}_{X;S} = \tilde{\ell}_{P^{\mathsf{M}}_{X,S,Y}}$, where $P^{\mathsf{M}}_{X,S,Y} \triangleq P_{X|S}P_SP_{Y|S}$. More generally, the space $\mathcal{F}_{\mathcal{X}\times\mathcal{S}}$ characterizes CDK functions associated with such Markov distributions, which we formalize as follows. A proof of which is provided in Appendix C.10.

Proposition 23 *Given* $Q_{X,S,Y}$ *with* $Q_X = P_X, Q_{S,Y} = P_{S,Y}$, *let* $\mathfrak{i}^{(Q)}_{X;S,Y}$ *denote the corresponding CDK function. Then,* $\mathfrak{i}^{(Q)}_{X;S,Y} \in \mathcal{F}_{\mathcal{X}\times\mathcal{S}}$ *if and only if* $Q_{X,S,Y} = Q_{X|S}Q_SQ_{Y|S}$.

Hence, we refer to the dependence component $\pi_{\mathsf{M}}\big(\mathfrak{i}_{X;S,Y}\big) = \mathfrak{i}_{X;S}$ as the Markov component. Then, we have $\pi_{\mathsf{C}}\big(\mathfrak{i}_{X;S,Y}\big) = \mathfrak{i}_{X;S,Y} - \mathfrak{i}_{X;S}$, which characterizes the joint dependence not captured by S. We refer to it as the Conditional dependence component, and also denote it by $\mathfrak{i}_{X;Y|S}$, i.e.,

$$\mathfrak{i}_{X;Y|S}(x,s,y) \triangleq \mathfrak{i}_{X;S,Y}(x,s,y) - \mathfrak{i}_{X;S}(x,s) = \left[\frac{P_{X,S,Y} - P^{\mathsf{M}}_{X,S,Y}}{R_{X,S,Y}}\right](x,s,y). \tag{55}$$

Therefore, the conditional dependence component $\mathfrak{i}_{X;Y|S}$ vanishes, i.e., $\|\mathfrak{i}_{X;Y|S}\| = 0$, if and only if X and Y are conditionally independent given S. In general, from the Pythagorean relation, we can write

$$\big\|\mathfrak{i}_{X;Y|S}\big\|^2 = \|\mathfrak{i}_{X;S,Y}\|^2 - \|\mathfrak{i}_{X;S}\|^2, \tag{56}$$

analogous to the expression of the conditional mutual information $I(X;Y|S) = I(X;S,Y) - I(X;S)$. Indeed, we can establish an explicit connection in the local regime where X and (S,Y) are ϵ-dependent, i.e., $\big\|\mathfrak{i}_{X;S,Y}\big\| = O(\epsilon)$. Then, from Lemma 9 we obtain $\big\|\mathfrak{i}_{X;S,Y}\big\|^2 = 2\cdot I(X;S,Y) + o(\epsilon^2)$, and similarly, $\big\|\mathfrak{i}_{X;S}\big\|^2 = 2\cdot I(X;S) + o(\epsilon^2)$. Therefore, (56) becomes

$$\|\mathfrak{i}_{X;Y|S}\|^2 = \big\|\mathfrak{i}_{X;S,Y}\big\|^2 - \big\|\mathfrak{i}_{X;S}\big\|^2 = 2\cdot I(X;Y|S) + o(\epsilon^2).$$

We then discuss learning these two dependence components by applying the nesting technique. To begin, note that since

$$\mathfrak{i}_{X;S} = \pi_{\mathsf{M}}(\mathfrak{i}_{X;S,Y}) = \Pi\left(\mathfrak{i}_{X;S,Y}; \mathcal{F}_{\mathcal{X}} \otimes \mathcal{F}_{\mathcal{S}}\right), \tag{57}$$

$$\mathfrak{i}_{X;Y|S} = \pi_{\mathsf{C}}(\mathfrak{i}_{X;S,Y}) = \Pi\left(\mathfrak{i}_{X;S,Y}; \mathcal{F}_{\mathcal{X}} \otimes \mathcal{F}_{\mathcal{Y}|\mathcal{S}}\right), \tag{58}$$

we recognize the decomposition $\mathfrak{i}_{X;S,Y} = \mathfrak{i}_{X;S} + \mathfrak{i}_{X;Y|S}$ as a special case of (43). Therefore, similar to our discussions in Section 4.3, we consider the nesting configuration $\mathcal{C}_{\mathsf{MC}}$ and its refinement $\mathcal{C}^{\star}_{\mathsf{MC}}$, where [cf. (44)]

$$\mathcal{C}_{\mathsf{MC}} \triangleq \left\{(\bar{k},k);\, \mathcal{F}_{\mathcal{X}};\, (\mathcal{F}_{\mathcal{S}}, \mathcal{F}_{\mathcal{S}\times\mathcal{Y}})\right\}. \tag{59}$$

The corresponding nested H-scores are defined on

$$\mathrm{dom}(\mathcal{C}_{\mathsf{MC}}) = \mathrm{dom}(\mathcal{C}^{\star}_{\mathsf{MC}}) = \left\{\left(\begin{bmatrix}\bar{f}\\ f\end{bmatrix}, \begin{bmatrix}\bar{g}\\ g\end{bmatrix}\right) : \bar{f} \in \mathcal{F}^{\bar{k}}_{\mathcal{X}}, \bar{g} \in \mathcal{F}^{\bar{k}}_{\mathcal{S}}, f \in \mathcal{F}^{k}_{\mathcal{X}}, g \in \mathcal{F}^{k}_{\mathcal{Y}\times\mathcal{S}}\right\}. \tag{60}$$

In particular, we can compute the nested H-score configured by $\mathcal{C}_{\mathsf{MC}}$ from a nested network structure as shown in Figure 10. Then, we can obtain both dependence components by optimizing the nested H-scores. Formally, we have the following corollary of Theorem 19 and Theorem 20.

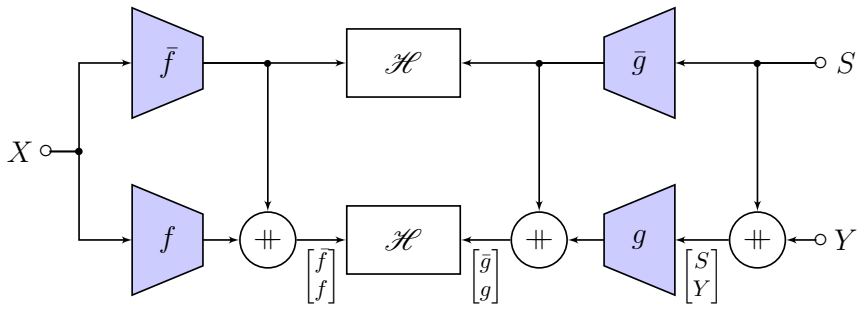


Figure 10: Nesting Technique for Learning With Side Information S

Corollary 24 *Given* $\bar{k} \geq \operatorname{rank}(\mathfrak{i}_{X;S})$, $\mathscr{H}\left(\begin{bmatrix}\bar{f}\\ f\end{bmatrix}, \begin{bmatrix}\bar{g}\\ g\end{bmatrix}; \mathsf{C}_{\mathsf{MC}}\right)$ *is maximized if and only if*

$$\bar{f} \otimes \bar{g} = \mathfrak{i}_{X;S}, \tag{61a}$$

$$f \otimes g = \zeta_{\leq k}(\mathfrak{i}_{X;Y|S}). \tag{61b}$$

In addition, $\mathscr{H}\left(\begin{bmatrix}\bar{f}\\ f\end{bmatrix}, \begin{bmatrix}\bar{g}\\ g\end{bmatrix}; \mathsf{C}^{\star}_{\mathsf{MC}}\right)$ *is maximized if and only if*

$$\bar{f}_i \otimes \bar{g}_i = \zeta_i(\mathfrak{i}_{X;S}), \quad i \in [\bar{k}]. \tag{62a}$$

$$f_i \otimes g_i = \zeta_i(\mathfrak{i}_{X;Y|S}), \quad i \in [k]. \tag{62b}$$

5.2 Feature Assembling and Inference Models

We then assemble the features for inference tasks, particularly the inference conditioned on S. We first consider the case where we have learned both dependence components $\mathfrak{i}_{X;S}$ and $\mathfrak{i}_{X;Y|S}$, for which we have the following characterization (cf. Proposition 12). A proof is provided in Appendix C.11.

Proposition 25 *Suppose features* $\bar{f} \in \mathcal{F}_{\mathcal{X}}^{\bar{k}}, \bar{g} \in \mathcal{F}_{\mathcal{S}}^{\bar{k}}$ *and* $f \in \mathcal{F}_{\mathcal{X}}^{k}, g \in \mathcal{F}_{\mathcal{S}\times\mathcal{Y}}^{k}$ *satisfy* $\bar{f}\otimes\bar{g} = \mathfrak{i}_{X;S}, f\otimes g = \mathfrak{i}_{X;Y|S}$. *Then, we have* $\|\mathfrak{i}_{X;S}\|^2 = \operatorname{tr}(\Lambda_{\bar{f}}\Lambda_{\bar{g}}), \|\mathfrak{i}_{X;Y|S}\|^2 = \operatorname{tr}(\Lambda_f\Lambda_g)$, *and*

$$P_{Y|X,S}(y|x,s) = P_{Y|S}(y|s) \cdot \left(1 + \frac{f^{\mathrm{T}}(x)g(s,y)}{1 + \bar{f}^{\,\mathrm{T}}(x)\bar{g}(s)}\right). \tag{63}$$

In addition, for any function $\psi \in \mathcal{F}_{\mathcal{Y}}^{d}$,

$$\mathbb{E}\left[\psi(Y)|X=x, S=s\right] = \mathbb{E}\left[\psi(Y)|S=s\right] + \frac{1}{1 + \bar{f}^{\,\mathrm{T}}(x)\bar{g}(s)} \cdot \Lambda_{\psi,g}^{(s)} f(x), \tag{64}$$

where we have defined $\Lambda_{\psi,g}^{(s)} \triangleq \mathbb{E}\left[\psi(Y)g^{\mathrm{T}}(s,Y)|S=s\right]$ *for each* $s \in \mathcal{S}$.

Therefore, we can compute the strength of both the Markov component $\mathfrak{i}_{X;S}$ and the conditional component $\mathfrak{i}_{X;Y|S}$ from the features. Similarly, we can further compute the spectrum of the dependence components, by learning the modal decomposition according to (62).

From Proposition 25, we can obtain inference models conditioned on the side information S. In particular, for classification task, we can use (63) to compute the posterior probability, with the resulting MAP estimation conditioned on $S = s$ [cf. (27)]:

$$\hat{y}_{\mathrm{MAP}}(x;s) = \arg\max_{y\in\mathcal{Y}} P_{Y|X,S}(y|x,s) = \arg\max_{y\in\mathcal{Y}} P_{Y|S}(y|s) \cdot \left(1 + \frac{f^{\mathrm{T}}(x)g(s,y)}{1 + \bar{f}^{\,\mathrm{T}}(x)\bar{g}(s)}\right). \tag{65}$$

Specifically, $P_{Y|S}$ can be obtained by a separate discriminative model that predicts Y from the side information S. In addition, when Y is continuous, we can obtain the MMSE estimator of $\psi(Y)$ conditioned on $S = s$ from (64), where we can learn $\mathbb{E}[\psi(Y)|S = s]$ and $\Lambda_{\psi,g}^{(s)} = \mathbb{E}\left[\psi(Y)g^{\mathrm{T}}(s, Y)|S = s\right]$ separately from (S, Y) pairs. As we construct both models by assembling learned features, the model outputs depend on input data X only through the features $\bar{f}$ and f of X, as desired.

Moreover, we can conduct a conditional independence test without learning the complete conditional dependence $\mathfrak{i}_{X;Y|S}$. In particular, suppose we have learned features $f \in \mathcal{F}_{\mathcal{X}}^k, g \in \mathcal{F}_{\mathcal{S}\times\mathcal{Y}}^k$ with $f \otimes g = \zeta_{\le k}(\mathfrak{i}_{X;Y|S})$ for some $k \ge 1$. Then we obtain $\mathrm{tr}(\Lambda_f\Lambda_g) = \|\zeta_{\le k}(\mathfrak{i}_{X;Y|S})\|^2 \ge 0$, where the equality holds if and only if $\mathfrak{i}_{X;Y|S} = 0$, i.e., X and Y are conditionally independent given S.

5.3 Theoretical Properties and Interpretations

We conclude this section by demonstrating theoretical properties of the learned features. In particular, we focus on the conditional dependence component $\mathfrak{i}_{X;Y|S}$ and associated features, as the Markov component $\mathfrak{i}_{X;S}$ shares the same properties as discussed in the bivariate case.

To begin, let $K \triangleq \mathrm{rank}(\mathfrak{i}_{X;Y|S})$, and let the modal decomposition of $\mathfrak{i}_{X;Y|S}$ be

$$\zeta_i(\mathfrak{i}_{X;Y|S}) = \sigma_i \cdot (f_i^* \otimes g_i^*), i \in [K], \tag{66}$$

where we have represented each mode in the standard form.

Then, we can interpret the σ_i, f_i^*, g_i^* as the solution to a constrained maximal correlation problem. To see this, note that from $\mathfrak{i}_{X;Y|S} = \Pi\left(\mathfrak{i}_{X;S,Y}; \mathcal{F}_{\mathcal{Y}|\mathcal{X},\mathcal{S}}\right) = \Pi\left(\mathfrak{i}_{X;S,Y}; \mathcal{F}_{\mathcal{X}} \otimes \mathcal{F}_{\mathcal{Y}|\mathcal{S}}\right)$, we can obtain $\sigma_i(f_i^* \otimes g_i^*) = \zeta_i(\mathfrak{i}_{X;Y|S}) = \zeta_i(\mathfrak{i}_{X;S,Y}|\mathcal{F}_{\mathcal{X}}, \mathcal{F}_{\mathcal{Y}|\mathcal{S}})$. Therefore, f_i^*, g_i^* are the constrained maximal correlation function of X and (S, Y) as defined in Proposition 7, with the subspaces $\mathcal{G}_{\mathcal{X}} = \mathcal{F}_{\mathcal{X}}, \mathcal{G}_{\mathcal{S}\times\mathcal{Y}} = \mathcal{F}_{\mathcal{Y}|\mathcal{S}}$.

5.3.1 Local Posterior Distribution and Conditional Dependence

In a local analysis regime, we can simplify the posterior distribution $P_{Y|X,S}$ as follows. A proof is provided in Appendix C.12.

Proposition 26 *If X and (S, Y) are ϵ-dependent, we have*

$$P_{Y|X,S}(y|x, s) = P_{Y|S}(y|s)\left(1 + \sum_{i=1}^{K} \sigma_i f_i^*(x) g_i^*(s, y)\right) + o(\epsilon), \tag{67}$$

where σ_i, f_i^*, g_i^* are as defined in (66).

From (67), the dominant term of $P_{Y|X,S}(y|x, s)$ depends on x only through $f_i^*(x)$, $i = 1, \ldots, K$. Therefore, the feature $f_{[K]}^*(X) = (f_1^*(X), \ldots, f_K^*(X))^{\mathrm{T}}$ captures the conditional dependence between X and Y given S except possibly higher-order terms of ϵ.

5.3.2 Relationship to Multitask Classification DNNs

We can also establish a connection between the side information problem and deep neural networks for multitask learning. Specifically, we consider a multitask classification task where X and Y denote the input data and target label to predict, respectively, and S denotes the index for tasks. When conditioned on different values of S, the dependence between data and label are generally different. We then demonstrate that a multitask DNN also learns the optimal approximation of the conditional dependence component $\mathfrak{i}_{X;Y|S}$.

We consider a classical multitask classification DNN design (Caruana, 1993; Ruder, 2017), as shown in Figure 11. In this figure, feature $f \in \mathcal{F}_{\mathcal{X}}^k$ of X is shared among all tasks. For each task

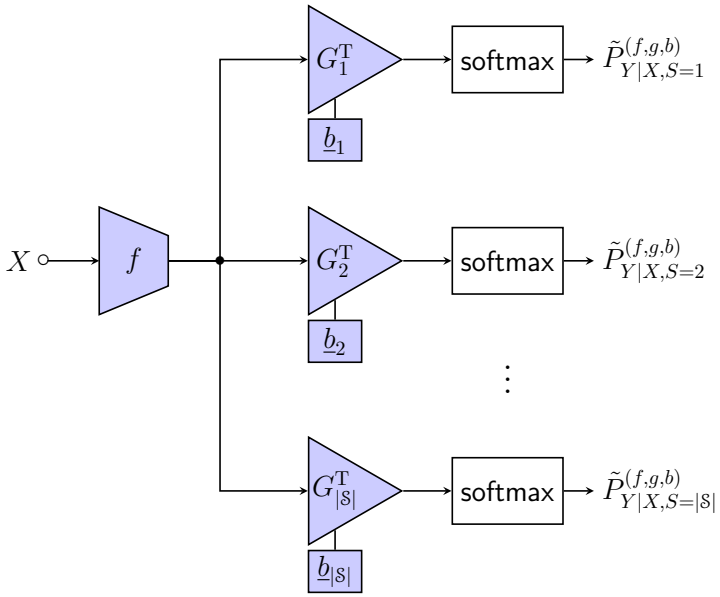


Figure 11: A multihead network for extracting feature f shared among different tasks in $\mathcal{S} = \{1, \ldots, |\mathcal{S}|\}$. Each task $s \in \mathcal{S}$ corresponds to a separate classification head with weight matrix G_s^{T} and bias vector $\underline{b}_s$ for generating the associated posterior $\tilde{P}^{(f,g,b)}_{Y|X,S=s}$.

$s \in \mathcal{S}$, the corresponding classification head with weight matrix $G_s^{\mathrm{T}} \in \mathbb{R}^{|\mathcal{Y}| \times k}$ and bias $\underline{b}_s \in \mathbb{R}^{|\mathcal{Y}|}$ are applied to compute the corresponding posterior probability

$$\tilde{P}^{(f,g,b)}_{Y|X,S}(y|x,s) \triangleq \frac{\exp\left(f(x) \cdot g(s,y) + b(s,y)\right)}{\sum_{y' \in \mathcal{Y}} \exp\left(f(x) \cdot g(s,y') + b(s,y')\right)}, \tag{68}$$

where $g \in \mathcal{F}_{\mathcal{S} \times \mathcal{Y}}$ and $b \in \mathcal{F}_{\mathcal{Y}}$ are related to G_s and $\underline{b}_s$ via [cf. (30)]

$$G_s(i,y) = g_i(s,y) \text{ for all } i \in [k], y \in \mathcal{Y}, \qquad \underline{b}_s = [b(s,1), \ldots, b(s,|\mathcal{Y}|)]^{\mathrm{T}}. \tag{69}$$

Given data samples $\{(x_i, s_i, y_i)\}_{i=1}^n$ with the empirical distribution $P_{X,S,Y}$, we write the corresponding likelihood function as

$$\mathcal{L}_S(f,g,b) \triangleq \frac{1}{n} \sum_{i=1}^n \log \tilde{P}^{(f,g,b)}_{Y|X,S}(y_i|x_i,s_i) = \mathbb{E}_{(\hat{X},\hat{Y},\hat{S}) \sim P_{X,Y,S}} \left[\log \tilde{P}^{(f,g,b)}_{Y|X,S}(\hat{Y}|\hat{X},\hat{S})\right]. \tag{70}$$

Note that we can relate the posterior probability $\tilde{P}^{(f,g,b)}_{Y|X,S}$ to the posterior $\tilde{P}^{(f,g,b)}_{Y|X}$ in the ordinary classification DNN, as defined in (31). To see this, note that for all $s \in \mathcal{S}$, we have $\tilde{P}^{(f,g,b)}_{Y|X,S=s} = \tilde{P}^{(f,g^{(s)},b^{(s)})}_{Y|X}$, where we have defined $g^{(s)} \in \mathcal{F}_{\mathcal{Y}}^k$ and $b^{(s)} \in \mathcal{F}_{\mathcal{Y}}$ for each $s \in \mathcal{S}$, as $g^{(s)}(y) \triangleq g(s,y), b^{(s)}(y) \triangleq b(s,y)$. Then, we rewrite (70) as $\mathcal{L}_S(f,g,b) = \sum_{s \in \mathcal{S}} P_S(s) \mathcal{L}_S^{(s)}(f, g^{(s)}, b^{(s)})$, where $\mathcal{L}_S^{(s)}(f,g,b) \triangleq \mathbb{E}_{(\hat{X},\hat{Y}) \sim P_{X,Y|S=s}} \left[\log \tilde{P}^{(f,g,b)}_{Y|X}(\hat{Y}|\hat{X})\right]$ is the expected likelihood value conditioned on $S = s$. We further assume the all bias terms are trained to their optimal values with respect to f and g. This gives the likelihood

$$\mathcal{L}_S(f,g) \triangleq \sum_{s \in \mathcal{S}} P_S(s) \mathcal{L}_S^{(s)}(f, g^{(s)}) = \max_{b \in \mathcal{F}_{\mathcal{S} \times \mathcal{Y}}} \mathcal{L}_S(f,g,b), \tag{71}$$

where we have denoted $\mathcal{L}_S^{(s)}(f,g) \triangleq \max_{b\in\mathcal{F}_\mathcal{Y}} \mathcal{L}_S^{(s)}(f,g,b)$ for each $s \in \mathcal{S}$. Then, from Property 3 we can verify that $\mathcal{L}_S(f,g)$ depends only on some centered features, formalized as follows.

Property 4 *We have* $\mathcal{L}_S(f,g) = \mathcal{L}_S(\tilde{f},\tilde{g})$, *where we have defined* $\tilde{f} \triangleq \Pi\left(f;\tilde{\mathcal{F}}_\mathcal{X}\right)$, *and* $\tilde{g} \triangleq \Pi\left(g;\mathcal{F}_{\mathcal{Y}|\mathcal{S}}\right)$, *i.e.,* $\tilde{f}(x) = f(x) - \mathbb{E}\left[f(X)\right]$ *and* $\tilde{g}(s,y) = g(s,y) - \mathbb{E}\left[g(s,Y)|S=s\right]$.

Therefore, we can focus on centered features $f \in \tilde{\mathcal{F}}_\mathcal{X}^k$ and $g \in \mathcal{F}_{\mathcal{Y}|\mathcal{S}}^k$, i.e., $\mathbb{E}\left[f(X)\right] = 0$ and $\mathbb{E}\left[g(s,Y)|S=s\right] = 0$ for all $s \in \mathcal{S}$. We also restrict to features f,g that perform better than the trivial choice of zero features, by assuming that

$$\mathcal{L}_S^{(s)}(f,g^{(s)}) \geq \mathcal{L}_S^{(s)}(0,g^{(s)}) = \mathcal{L}_S^{(s)}(0,0) = -H(Y|S=s), \quad \text{for all } s \in \mathcal{S}. \tag{72}$$

Then, we have the following characterization, which extends Proposition 14 to the multitask setting. A proof is provided in Appendix C.13.

Theorem 27 *Suppose* X *and* (S,Y) *are* ϵ*-dependent. For* $f \in \tilde{\mathcal{F}}_\mathcal{X}^k$ *and* $g \in \mathcal{F}_{\mathcal{Y}|\mathcal{S}}^k$ *with* (72),

$$\mathcal{L}_S(f,g) = \mathcal{L}_S(0,0) + \frac{1}{2}\cdot\left(\|\mathsf{i}_{X;Y|S}\|^2 - \left\|\mathsf{i}_{X;Y|S} - f\otimes g\right\|^2\right) + o(\epsilon^2), \tag{73}$$

which is maximized if and only if $f\otimes g = \zeta_{\leq k}(\mathsf{i}_{X;Y|S}) + o(\epsilon)$.

Therefore, the multitask classification network essentially learns features approximating the conditional dependence $\mathsf{i}_{X;Y|S}$. Different from the nested H-score implementation, this multitask network implements the conditioning by directly applying a separate classification head for each task $S=s$. As a consequence, this design requires $|\mathcal{S}|$ many different heads, and is not applicable when the side information S is continuous or has complicated structures.

6. Multimodal Learning With Missing Modalities

In this section, we demonstrate another multivariate learning application, where we need to conduct inferences based on different data sources. In particular, we focus on the setting where the goal is to infer Y from two different data sources, denoted by X_1 and X_2.

We refer to such problems as the multimodal learning[12] problems, and are particularly interested in the cases where we have missing modalities: either X_1 or X_2 can be missing during the inference. Our goal is to design a learning system to solve all the three problems: (i) inferring Y based on X_1, (ii) inferring Y based on X_2, and (iii) inferring Y based on (X_1,X_2).

Throughout our discussions in this section, we use $P_{X_1,X_2,Y}$ to denote the joint distribution of $X_1 \in \mathcal{X}_1, X_2 \in \mathcal{X}_2, Y \in \mathcal{Y}$. For convenience, we also denote $X \triangleq (X_1,X_2) \in \mathcal{X} \triangleq \mathcal{X}_1\times\mathcal{X}_2$. We consider the feature geometry on $\mathcal{F}_{\mathcal{X}\times\mathcal{Y}} = \mathcal{F}_{\mathcal{X}_1\times\mathcal{X}_2\times\mathcal{Y}}$, with the metric distribution $R_{X_1,X_2,Y} = P_{X_1,X_2}P_Y$, or equivalently, $R_{X,Y} = P_XP_Y$.

6.1 Dependence Decomposition

To begin, we decompose the joint dependence $\mathsf{i}_{X_1,X_2;Y} \in \mathcal{F}_{\mathcal{X}_1\times\mathcal{X}_2\times\mathcal{Y}}$ as

$$\mathsf{i}_{X_1,X_2;Y} = \pi_{\mathsf{B}}(\mathsf{i}_{X_1,X_2;Y}) + \pi_{\mathsf{I}}(\mathsf{i}_{X_1,X_2;Y}), \tag{74}$$

12. The literature typically uses "multimodal" to refer to the different forms (modalities) of data sources, e.g., video, audio, and text. However, such distinction is insignificant in our treatment, when we model each data source as a random variable.

where we have defined $\pi_{\mathsf{B}}(\gamma) \triangleq \Pi\left(\gamma; \mathcal{F}_{\mathcal{X}_1\times\mathcal{Y}} + \mathcal{F}_{\mathcal{X}_2\times\mathcal{Y}}\right), \pi_{\mathsf{I}}(\gamma) \triangleq \gamma - \pi_{\mathsf{B}}(\gamma)$. We refer to $\pi_{\mathsf{B}}(\mathrm{i}_{X_1,X_2;Y})$ as the Bivariate dependence component, and refer to $\pi_{\mathsf{I}}(\mathrm{i}_{X_1,X_2;Y})$ as the Interaction component.

The bivariate dependence component $\pi_{\mathsf{B}}(\mathrm{i}_{X_1,X_2;Y})$ is uniquely determined by all pairwise dependencies among X_1, X_2, Y. Formally, let $\mathcal{Q}_{\mathsf{B}}$ denote the collection of distributions with the same pairwise marginal distributions as $P_{X_1,X_2,Y}$, i.e.,

$$\mathcal{Q}_{\mathsf{B}} \triangleq \{Q_{X_1,X_2,Y} \in \mathcal{P}^{\mathcal{X}_1\times\mathcal{X}_2\times\mathcal{Y}} \colon Q_{X_1,X_2} = P_{X_1,X_2}, Q_{X_1,Y} = P_{X_1,Y}, Q_{X_2,Y} = P_{X_2,Y}\}. \tag{75}$$

Then we have the following result. A proof is provided in Appendix C.14.

Proposition 28 *For all $Q_{X_1,X_2,Y} \in \mathcal{Q}_{\mathsf{B}}$, we have $\pi_{\mathsf{B}}(\mathrm{i}^{(Q)}_{X_1,X_2;Y}) = \pi_{\mathsf{B}}(\mathrm{i}_{X_1,X_2;Y})$, where $\mathrm{i}^{(Q)}_{X_1,X_2;Y}$ denotes the CDK function associated with $Q_{X_1,X_2,Y}$.*

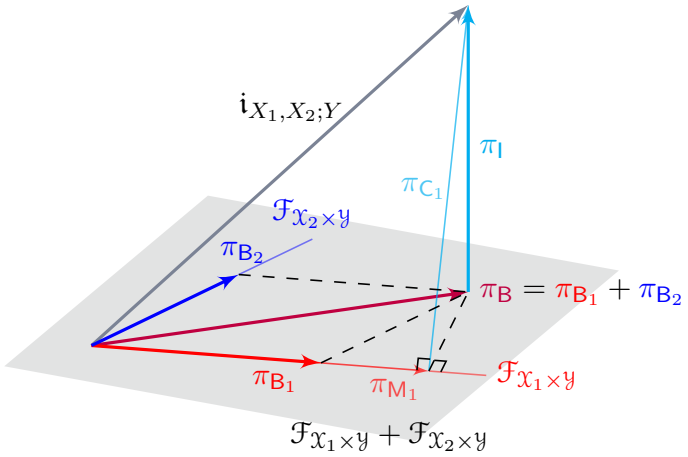


Figure 12: Decompose the joint dependence $\mathrm{i}_{X_1,X_2;Y}$ into bivariate dependence component π_{B} and interaction dependence component π_{I}. The plane denotes the sum of $\mathcal{F}_{\mathcal{X}_1\times\mathcal{Y}}$ and $\mathcal{F}_{\mathcal{X}_2\times\mathcal{Y}}$.

We show the relation between different dependence components in Figure 12, where we have further decomposed the bivariate dependence component $\pi_{\mathsf{B}}(\mathrm{i}_{X_1,X_2;Y})$ as $\pi_{\mathsf{B}}(\mathrm{i}_{X_1,X_2;Y}) = \pi_{\mathsf{B}_1}(\mathrm{i}_{X_1,X_2;Y}) + \pi_{\mathsf{B}_2}(\mathrm{i}_{X_1,X_2;Y})$ for some $\pi_{\mathsf{B}_i}(\mathrm{i}_{X_1,X_2;Y}) \in \mathcal{F}_{\mathcal{X}_i\times\mathcal{Y}}, i = 1, 2$. For comparison, we have also demonstrated $\pi_{\mathsf{M}_1}(\mathrm{i}_{X_1,X_2;Y}) = \mathrm{i}_{X_1;Y}$ and $\pi_{\mathsf{C}_1}(\mathrm{i}_{X_1,X_2;Y}) = \mathrm{i}_{X_1,X_2;Y} - \mathrm{i}_{X_1;Y}$, obtained from the decomposition introduced in Section 5.1. Note that since the interaction component $\pi_{\mathsf{I}}(\mathrm{i}_{X_1;X_2;Y})$ does not capture any bivariate dependence, we can also obtain $\pi_{\mathsf{M}_1}(\mathrm{i}_{X_1,X_2;Y})$ directly from the bivariate component $\pi_{\mathsf{B}}(\mathrm{i}_{X_1,X_2;Y})$ via a projection: $\pi_{\mathsf{M}_1}(\mathrm{i}_{X_1,X_2;Y}) = \pi_{\mathsf{M}_1}(\pi_{\mathsf{B}}(\mathrm{i}_{X_1,X_2;Y}))$.

6.2 Feature Learning from Complete Data

We consider learning the features representations for the two dependence components. Here, we assume the data are complete (X_1, X_2, Y) triplets with the empirical distribution $P_{X_1,X_2,Y}$. We will discuss the learning with incomplete data later.

Again, we apply the nesting technique to design the training objective. Note that since $X = (X_1, X_2)$, with $\mathcal{G}_{\mathcal{X}} = \mathcal{F}_{\mathcal{X}_1} + \mathcal{F}_{\mathcal{X}_2}$, we can express the two components as [cf. (43)]

$$\pi_{\mathsf{B}}(\mathrm{i}_{X;Y}) = \Pi\left(\mathrm{i}_{X;Y}; \mathcal{G}_{\mathcal{X}} \otimes \mathcal{F}_{\mathcal{Y}}\right), \tag{76}$$

$$\pi_{\mathsf{I}}(\mathrm{i}_{X;Y}) = \Pi\left(\mathrm{i}_{X;Y}; (\mathcal{F}_{\mathcal{X}} \boxminus \mathcal{G}_{\mathcal{X}}) \otimes \mathcal{F}_{\mathcal{Y}}\right). \tag{77}$$

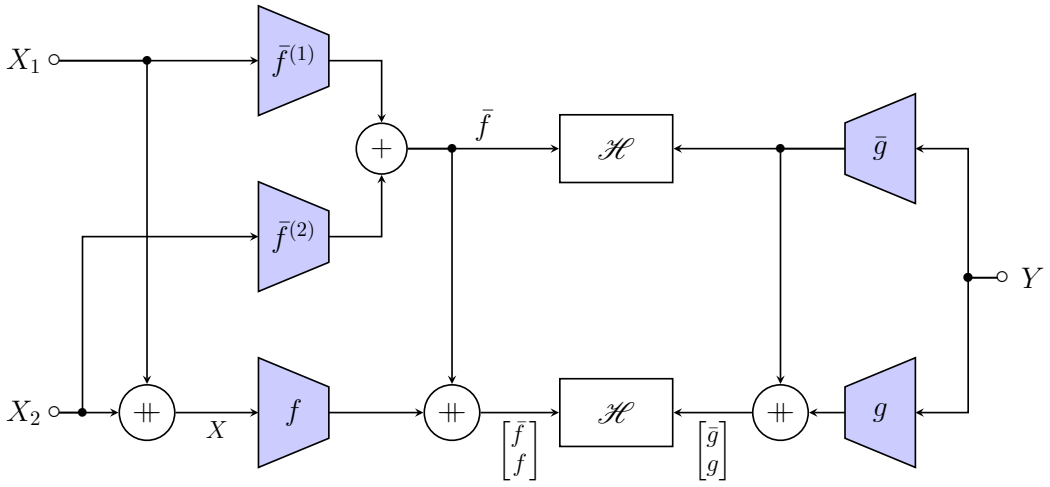


Figure 13: Nesting Technique for Learning Features from Multimodal Data

Therefore, we consider the nesting configuration $\mathcal{C}_{\mathsf{BI}}$ and its refinement $\mathcal{C}^{\star}_{\mathsf{BI}}$, as [cf. (44)]

$$\mathcal{C}_{\mathsf{BI}} \triangleq \left\{(\bar{k}, k);\, (\mathcal{F}_{\mathcal{X}_1} + \mathcal{F}_{\mathcal{X}_2}, \mathcal{F}_{\mathcal{X}});\, \mathcal{F}_{\mathcal{Y}}\right\}. \tag{78}$$

The corresponding nested H-scores are defined on

$$\mathrm{dom}(\mathcal{C}_{\mathsf{BI}}) = \mathrm{dom}(\mathcal{C}^{\star}_{\mathsf{BI}}) = \left\{\left(\begin{bmatrix}\bar{f}\\ f\end{bmatrix}, \begin{bmatrix}\bar{g}\\ g\end{bmatrix}\right) : \bar{f} \in \mathcal{F}^{\bar{k}}_{\mathcal{X}_1} + \mathcal{F}^{\bar{k}}_{\mathcal{X}_2}, \bar{g} \in \mathcal{F}^{\bar{k}}_{\mathcal{Y}}, f \in \mathcal{F}^{k}_{\mathcal{X}}, g \in \mathcal{F}^{k}_{\mathcal{Y}}\right\}. \tag{79}$$

Specifically, we can compute the nested H-score configured by $\mathcal{C}_{\mathsf{BI}}$ using a nested network structure as shown in Figure 13. Then we can obtain both dependence components by maximizing the corresponding nested H-scores, formalized as follows (cf. Theorem 19 and Theorem 20).

Corollary 29 *Given $\bar{k} \geq \mathrm{rank}(\pi_{\mathsf{B}}(\mathfrak{i}_{X_1,X_2;Y}))$, the nested H-score $\mathscr{H}\left(\begin{bmatrix}\bar{f}\\ f\end{bmatrix}, \begin{bmatrix}\bar{g}\\ g\end{bmatrix}; \mathcal{C}_{\mathsf{BI}}\right)$ is maximized if and only if*

$$\bar{f} \otimes \bar{g} = \pi_{\mathsf{B}}(\mathfrak{i}_{X_1,X_2;Y}), \tag{80a}$$

$$f \otimes g = \zeta_{\leq k}\left(\pi_{\mathsf{I}}(\mathfrak{i}_{X_1,X_2;Y})\right). \tag{80b}$$

In addition, $\mathscr{H}\left(\begin{bmatrix}\bar{f}\\ f\end{bmatrix}, \begin{bmatrix}\bar{g}\\ g\end{bmatrix}; \mathcal{C}^{\star}_{\mathsf{BI}}\right)$ is maximized if and only if

$$\bar{f}_i \otimes \bar{g}_i = \zeta_i\left(\pi_{\mathsf{B}}(\mathfrak{i}_{X_1,X_2;Y})\right), \quad i \in [\bar{k}], \tag{81a}$$

$$f_i \otimes g_i = \zeta_i\left(\pi_{\mathsf{I}}(\mathfrak{i}_{X_1,X_2;Y})\right), \quad i \in [k]. \tag{81b}$$

6.3 Feature Assembling and Inference Models

We then illustrate how to assemble the learned features for the inference tasks and deal with incomplete data. For convenience, we define the conditional expectation operators $\tau_i, i = 1, 2$, such that for $f \in \mathcal{F}^{k}_{\mathcal{X}_1 \times \mathcal{X}_2}$ with $k \geq 1$, we have

$$[\tau_i(f)](x_i) \triangleq \mathbb{E}\left[f(X_1, X_2) | X_i = x_i\right], \qquad \text{for all } x_i \in \mathcal{X}_i. \tag{82}$$

Note that we can also interpret τ_i as to the projection onto $\mathcal{F}_{\mathcal{X}_i}$, i.e., $\tau_i(f) = \Pi\left(f; \mathcal{F}_{\mathcal{X}_i}\right)$. Then, we have the following result. A proof is provided in Appendix C.15.

Proposition 30 *Suppose we have* $\bar{f} \otimes \bar{g} = \pi_{\mathsf{B}}(\mathfrak{i}_{X_1,X_2;Y}), f \otimes g = \pi_{\mathsf{I}}(\mathfrak{i}_{X_1,X_2;Y})$ *for features* $\bar{f} = \bar{f}^{(1)} + \bar{f}^{(2)}$ *with* $\bar{f}^{(i)} \in \mathcal{F}^{\bar{k}}_{\mathcal{X}_i}, i = 1, 2$, $\bar{g} \in \mathcal{F}^{\bar{k}}_{\mathcal{Y}}$, $f \in \mathcal{F}^{k}_{\mathcal{X}}$, *and* $g \in \mathcal{F}^{k}_{\mathcal{Y}}$. *Then, we have*

$$P_{Y|X_1,X_2}(y|x_1,x_2) = P_Y(y)\left[1 + \bar{f}^{\mathrm{T}}(x_1,x_2)\bar{g}(y) + f^{\mathrm{T}}(x_1,x_2)g(y)\right], \tag{83a}$$

$$P_{Y|X_1}(y|x_1) = P_Y(y)\left[1 + \left(\bar{f}^{(1)}(x_1) + [\tau_1(\bar{f}^{(2)})](x_1)\right)^{\mathrm{T}}\bar{g}(y)\right], \tag{83b}$$

$$P_{Y|X_2}(y|x_2) = P_Y(y)\left[1 + \left(\bar{f}^{(2)}(x_2) + [\tau_2(\bar{f}^{(1)})](x_2)\right)^{\mathrm{T}}\bar{g}(y)\right]. \tag{83c}$$

In addition, for all $\psi \in \mathcal{F}^{d}_{\mathcal{Y}}$, *we have*

$$\mathbb{E}\left[\psi(Y)|X_1 = x_1, X_2 = x_2\right] = \mathbb{E}\left[\psi(Y)\right] + \Lambda_{\psi,\bar{g}}\bar{f}(x_1,x_2) + \Lambda_{\psi,g}f(x_1,x_2), \tag{84a}$$

$$\mathbb{E}\left[\psi(Y)|X_1 = x_1\right] = \mathbb{E}\left[\psi(Y)\right] + \Lambda_{\psi,\bar{g}}\left(\bar{f}^{(1)}(x_1) + [\tau_1(\bar{f}^{(2)})](x_1)\right), \tag{84b}$$

$$\mathbb{E}\left[\psi(Y)|X_2 = x_2\right] = \mathbb{E}\left[\psi(Y)\right] + \Lambda_{\psi,\bar{g}}\left(\bar{f}^{(2)}(x_2) + [\tau_2(\bar{f}^{(1)})](x_2)\right). \tag{84c}$$

From Proposition 30, we can obtain inference models for all three different types of input data, by simply assembling the learned features in different ways. The resulting inference models also reveal the different roles of two dependence components. For example, the features associated with the interaction dependence component, i.e., f and g, are used only when we have both X_1 and X_2 observations.

In practice, we can use (83) and (84) for classification and estimation tasks, respectively. To apply (84), we can compute $\Lambda_{\psi,\bar{g}}$ and $\Lambda_{\psi,g}$ from the corresponding empirical averages over the training dataset, and learn features $\tau_1(\bar{f}^{(2)})$ and $\tau_2(\bar{f}^{(1)})$ from (X_1, X_2) pairs. For example, we can use Proposition 12 to implement the conditional expectation operators τ_1 and τ_2 [cf. (26)].

6.4 Theoretical Properties and Interpretations

We then introduce several theoretical properties of the dependence decomposition and induced feature representations, including their connections to the principle of maximum entropy (Jaynes, 1957a,b) and the optimal transformation of variables (Breiman and Friedman, 1985).

6.4.1 Dependence Decomposition

We can relate the bivariate-interaction decomposition (74) to decomposition operations in both the probability distribution space and the data space.

Decomposition in Distribution Space We assume that for all $(x_1, x_2, y) \in \mathcal{X}_1 \times \mathcal{X}_2 \times \mathcal{Y}$,

$$[\pi_{\mathsf{B}}(\mathfrak{i}_{X_1,X_2;Y})](x_1,x_2,y) \geq -1, \quad [\pi_{\mathsf{I}}(\mathfrak{i}_{X_1,X_2;Y})](x_1,x_2,y) \geq -1, \tag{85}$$

and define the associated distributions

$$P^{\mathsf{B}}_{X_1,X_2,Y}(x_1,x_2,y) \triangleq P_{X_1,X_2}(x_1,x_2)P_Y(y)(1 + [\pi_{\mathsf{B}}(\mathfrak{i}_{X_1,X_2;Y})](x_1,x_2,y)), \tag{86a}$$

$$P^{\mathsf{I}}_{X_1,X_2,Y}(x_1,x_2,y) \triangleq P_{X_1,X_2}(x_1,x_2)P_Y(y)(1 + [\pi_{\mathsf{I}}(\mathfrak{i}_{X_1,X_2;Y})](x_1,x_2,y)). \tag{86b}$$

Then, we have the following characterization, a proof of which is provided in Appendix C.16.

Proposition 31 *Under assumption* (85), *we have* $P^{\mathsf{B}}_{X_1,X_2,Y}, P^{\mathsf{I}}_{X_1,X_2,Y} \in \mathcal{P}^{\mathcal{X}_1\times\mathcal{X}_2\times\mathcal{Y}}$, *with marginal distributions* $P^{\mathsf{B}}_{X_1,X_2} = P^{\mathsf{I}}_{X_1,X_2} = P_{X_1,X_2}$ *and* $P^{\mathsf{B}}_{X_i,Y} = P_{X_i,Y}, P^{\mathsf{I}}_{X_i,Y} = P_{X_i}P_Y$ *for* $i = 1, 2$.

From Proposition 31, $P^{\mathsf{I}}_{X_1,X_2,Y}$ has marginal distributions $P^{\mathsf{I}}_{X_i,Y} = P_{X_i}P_Y$, $i = 1, 2$, and does not capture $(X_1; Y)$ or $(X_2; Y)$ dependence. On the other hand, $P^{\mathsf{B}}_{X_1,X_2,Y}$ has the same pairwise marginal distributions as $P_{X_1,X_2,Y}$, i.e., $P^{\mathsf{B}}_{X_1,X_2,Y} \in \mathcal{Q}_{\mathsf{B}}$ with $\mathcal{Q}_{\mathsf{B}}$ as defined in (75). We can show that $P^{\mathsf{B}}_{X_1,X_2,Y}$ also achieves the maximum entropy in $\mathcal{Q}_{\mathsf{B}}$ in the local analysis regime. Formally, let

$$P^{\mathsf{ent}}_{X_1,X_2,Y} \triangleq \operatorname*{arg\,max}_{Q_{X_1,X_2,Y} \in \mathcal{Q}_{\mathsf{B}}} H(Q_{X_1,X_2,Y}) \tag{87}$$

denote the entropy maximizing distribution on $\mathcal{Q}_{\mathsf{B}}$, where $H(Q_{X_1,X_2,Y})$ denotes the entropy of $(X_1, X_2, Y) \sim Q_{X_1,X_2,Y}$. Then we have the following result. A proof is provided in Appendix C.17.

Proposition 32 *Suppose $X = (X_1, X_2)$ and Y are ϵ-dependent, and let $\mathfrak{i}^{(\mathsf{ent})}_{X_1,X_2;Y}$ denote the CDK function associated with $P^{\mathsf{ent}}_{X_1,X_2,Y}$. Then, we have $\left\|\pi_{\mathsf{B}}(\mathfrak{i}_{X_1,X_2;Y}) - \mathfrak{i}^{(\mathsf{ent})}_{X_1,X_2;Y}\right\| = o(\epsilon)$, or equivalently,*

$$P^{\mathsf{ent}}_{X_1,X_2,Y}(x_1, x_2, y) = P^{\mathsf{B}}_{X_1,X_2,Y}(x_1, x_2, y) + o(\epsilon), \quad \text{for all } x_1, x_2, y. \tag{88}$$

Decomposition in Data Space For each triplet $(x_1, x_2, y) \in \mathcal{X}_1 \times \mathcal{X}_2 \times \mathcal{Y}$, we consider the decomposition

$$(x_1, x_2, y) \mapsto (x_1, x_2), (x_1, y), (x_2, y). \tag{89}$$

Suppose the dataset[13] $\mathscr{D} \triangleq \left\{\left(x_1^{(i)}, x_2^{(i)}, y^{(i)}\right)\right\}_{i\in[n]}$ has the empirical distribution $P_{X_1,X_2,Y}$, where each tuple $\left(x_1^{(i)}, x_2^{(i)}, y^{(i)}\right) \in \mathcal{X}_1 \times \mathcal{X}_2 \times \mathcal{Y}$. Then, by applying this decomposition on $\mathscr{D}$ and grouping the decomposed pairs, we obtain three separate datasets

$$\left\{\left(x_1^{(i)}, x_2^{(i)}\right)\right\}_{i\in[n]}, \left\{\left(x_1^{(i)}, y^{(i)}\right)\right\}_{i\in[n]}, \left\{\left(x_2^{(i)}, y^{(i)}\right)\right\}_{i\in[n]}, \tag{90}$$

which have empirical distributions P_{X_1,X_2}, $P_{X_1,Y}$, and $P_{X_2,Y}$, respectively.

Therefore, we can interpret the decomposition (89) as extracting the bivariate dependence component from the joint dependence: the new pairwise datasets retain all pairwise dependence, but do not capture any interaction among X_1, X_2, Y. Indeed, it is easy to see that, for each dataset with an empirical distribution taken from $\mathcal{Q}_{\mathsf{B}}$, the decomposition (89) leads to the same pairwise datasets. Reversely, we can reconstruct $P^{\mathsf{B}}_{X_1,X_2,Y}$ from the pairwise datasets (90). We will discuss the reconstruction algorithm design later.

6.4.2 Feature Representations

Let $\bar{K} \triangleq \operatorname{rank}(\pi_{\mathsf{B}}(\mathfrak{i}_{X_1,X_2;Y}))$ and $K \triangleq \operatorname{rank}(\pi_{\mathsf{I}}(\mathfrak{i}_{X_1,X_2;Y}))$. Then, we can represent the dependence modes of the bivariate component $\pi_{\mathsf{B}}(\mathfrak{i}_{X_1,X_2;Y})$ and $\pi_{\mathsf{I}}(\mathfrak{i}_{X_1,X_2;Y})$ in their standard forms, as

$$\zeta_i(\pi_{\mathsf{B}}(\mathfrak{i}_{X_1,X_2;Y})) = \bar{\sigma}_i\,(\bar{f}_i^* \otimes \bar{g}_i^*), \quad i \in [\bar{K}], \tag{91a}$$

$$\zeta_i(\pi_{\mathsf{I}}(\mathfrak{i}_{X_1,X_2;Y})) = \sigma_i\,(f_i^* \otimes g_i^*), \quad i \in [K]. \tag{91b}$$

By applying Proposition 7, we can interpret these features as solutions to corresponding constrained maximal correlation problems. For example, since $\zeta_i(\pi_{\mathsf{B}}(\mathfrak{i}_{X_1,X_2;Y})) = \zeta_i(\mathfrak{i}_{X_1,X_2;Y}|\mathcal{F}_{\mathcal{X}_1} + \mathcal{F}_{\mathcal{X}_2}, \mathcal{F}_{\mathcal{Y}})$, $(\bar{f}_i^*, \bar{g}_i^*)$ is the i-th constrained maximal correlation function pair of $X = (X_1, X_2)$ and Y restricted to subspaces $\mathcal{F}_{\mathcal{X}_1} + \mathcal{F}_{\mathcal{X}_2}$ and $\mathcal{F}_{\mathcal{Y}}$, respectively.

13. Though the dataset is modeled as a multiset without ordering, we introduce the index i for the convenience of presentation, which corresponds to a specific realization for traversing the dataset.

The top mode $(\bar{\sigma}_1, \bar{f}_1^*, \bar{g}_1^*)$ in (91a) also characterizes the optimal solution to a classical regression formulation. Specifically, given input variables X_1, X_2 and the output variable Y, Breiman and Friedman (1985) formulated the regression problem

$$\underset{\substack{\phi^{(1)}\in\tilde{\mathcal{F}}_{\mathcal{X}_1},\phi^{(2)}\in\tilde{\mathcal{F}}_{\mathcal{X}_2}\\ \psi\in\tilde{\mathcal{F}}_{\mathcal{Y}}\colon \|\psi\|=1}}{\text{minimize}} \mathbb{E}\left[\left(\psi(Y)-\phi^{(1)}(X_1)-\phi^{(2)}(X_2)\right)^2\right], \tag{92}$$

where the minimization is over zero-mean functions $\phi^{(1)}, \phi^{(2)}$, and ψ. The solution of (92), referred to as the optimal transformations (Breiman and Friedman, 1985), can be characterized as follows. A proof is provided in Appendix C.18.

Proposition 33 *The minimum value of optimization problem* (92) *is* $1-\bar{\sigma}_1^2$*, which can be achieved by* $\phi^{(1)}+\phi^{(2)} = \bar{\sigma}_1 \cdot \bar{f}_1^*$ *and* $\psi = \bar{g}_1^*$*.*

Therefore, the optimal transformations depend on, and thus characterize only, the top mode of the bivariate dependence component $\pi_{\mathsf{B}}(\mathfrak{i}_{X_1,X_2;Y})$.

6.5 Learning With Missing Modalities

We conclude this section by briefly discussing feature learning based on incomplete samples.

6.5.1 Learning from Pairwise Samples

A special case of the incomplete samples is the pairwise datasets (90) obtained from the decomposition (89). Specifically, suppose we obtain (90) from $\mathscr{D} \triangleq \left\{\left(x_1^{(i)}, x_2^{(i)}, y^{(i)}\right)\right\}_{i\in[n]}$, and let $P_{X_1,X_2,Y}$ denote the empirical distribution of $\mathscr{D}$. Since the bivariate dependence is retained in the decomposition (89), we can learn $\pi_{\mathsf{B}}(\mathfrak{i}_{X_1,X_2;Y})$ from the pairwise datasets (90).

In particular, when we set $k=0$ in $\mathcal{C}_{\mathsf{BI}}$ [cf. (78)], we have $\mathscr{H}\left(\begin{bmatrix}\bar{f}\\ f\end{bmatrix}, \begin{bmatrix}\bar{g}\\ g\end{bmatrix}; \mathcal{C}_{\mathsf{BI}}\right) = 2\cdot\mathscr{H}(\bar{f},\bar{g})$, and

$$\begin{aligned}
\mathscr{H}(\bar{f},\bar{g}) &= \mathscr{H}\left(\bar{f}^{(1)}+\bar{f}^{(2)}, \bar{g}\right)\\
&= \mathbb{E}\left[\left(\bar{f}^{(1)}(X_1)+\bar{f}^{(2)}(X_2)\right)^{\mathrm{T}}\bar{g}(Y)\right] - \left(\mathbb{E}\left[\bar{f}^{(1)}(X_1)+\bar{f}^{(2)}(X_2)\right]\right)^{\mathrm{T}}\mathbb{E}\left[\bar{g}(Y)\right] - \frac{1}{2}\operatorname{tr}\left(\Lambda_{\bar{f}}\Lambda_{\bar{g}}\right)\\
&= \mathscr{H}(\bar{f}^{(1)},\bar{g}) + \mathscr{H}(\bar{f}^{(2)},\bar{g}) - \operatorname{tr}\left(\Lambda_{\bar{f}^{(1)},\bar{f}^{(2)}}\Lambda_{\bar{g}}\right).
\end{aligned} \tag{93}$$

Therefore, we can evaluate (93) from the pairwise datasets (90), since each $\mathscr{H}(\bar{f}^{(i)},\bar{g})$ depends only on $P_{X_i,Y}$ for $i=1,2$, and $\Lambda_{\bar{f}^{(1)},\bar{f}^{(2)}}$ depends only on P_{X_1,X_2}. Then, from Corollary 29, we can obtain $\pi_{\mathsf{B}}(\mathfrak{i}_{X_1,X_2;Y})$ and the same set of features.

6.5.2 General Heterogeneous Training Data

We then consider general forms of heterogeneous training data, as shown in Table 1. In particular, suppose there are $n \triangleq n_0+n_1+n_2$ training samples, and we group them into separate datasets: $\mathscr{D}_0$ contains n_0 complete observations of (X_1,X_2,Y), and, for $i=1,2$, each $\mathscr{D}_i$ has n_i sample pairs of (X_i,Y). Our goal is to learn features from the heterogeneous data and obtain similar inference models as we introduced in Section 6.3.

In this case, in addition to the empirical distributions of these datasets, we also need to consider the sample sizes n_0, n_1, n_2 that indicate the relative qualities. To begin, we use a metric distribution

Datasets	Empirical Distribution	Remark
$\mathscr{D}_0 = \{(x_1^{(i)}, x_2^{(i)}, y^{(i)})\}_{i=1}^{n_0}$	$\hat{P}^{(0)}_{X_1,X_2,Y}$	Complete Observation
$\mathscr{D}_1 = \{(x_1^{(i)}, y^{(i)})\}_{i=n_0+1}^{n_0+n_1}$	$\hat{P}^{(1)}_{X_1,Y}$	X_2 missing
$\mathscr{D}_2 = \{(x_2^{(i)}, y^{(i)})\}_{i=n_0+n_1+1}^{n_0+n_1+n_2}$	$\hat{P}^{(2)}_{X_2,Y}$	X_1 missing

Table 1: Heterogeneous Training Data With Missing Modalities

of the product form $R_{X_1,X_2,Y} = R_{X_1,X_2}R_Y$, where R_{X_1,X_2} and R_Y correspond to some empirical distributions of training data. For example, we can set $R_{X_1,X_2} = \hat{P}^{(0)}_{X_1,X_2}$ and $R_Y = \eta_0\hat{P}^{(0)}_Y + \eta_1\hat{P}^{(1)}_Y + \eta_2\hat{P}^{(2)}_Y$ with $\eta_i \triangleq n_i/n$ for $i = 0, 1, 2$, which correspond to the empirical distributions of all (X_1, X_2) sample pairs and all Y samples, respectively.

Then, for any given $Q_{X_1,X_2,Y} \in \mathcal{P}^{\mathcal{X}_1\times\mathcal{X}_2\times Y}$, we use a weighted sum $L(Q_{X_1,X_2,Y})$ to characterize the difference between $Q_{X_1,X_2,Y}$ and the heterogeneous observations, defined as

$$L(Q_{X_1,X_2,Y}) \triangleq \eta_0 \cdot \big\|\tilde{\ell}_{\hat{P}^{(0)}_{X_1,X_2,Y}} - \tilde{\ell}_{Q_{X_1,X_2,Y}}\big\|^2 + \eta_1 \cdot \big\|\tilde{\ell}_{\hat{P}^{(1)}_{X_1,Y}} - \tilde{\ell}_{Q_{X_1,Y}}\big\|^2 + \eta_2 \cdot \big\|\tilde{\ell}_{\hat{P}^{(2)}_{X_2,Y}} - \tilde{\ell}_{Q_{X_2,Y}}\big\|^2. \tag{94}$$

We use $P^{(\text{est})}_{X_1,X_2,Y}$ to denote the optimal distribution that minimizes (94).

We can again apply the nesting technique to learn the feature representations associated with $P^{(\text{est})}_{X_1,X_2,Y}$. To begin, we use $\mathscr{H}(f, g; Q_{X,Y})$ to denote the H-score computed over the joint distribution $Q_{X,Y}$, defined as

$$\begin{aligned}\mathscr{H}(f, g; Q_{X,Y}) &\triangleq \frac{1}{2}\left(\big\|\tilde{\ell}_{Q_{X,Y}}\big\|^2 - \Big\|\tilde{\ell}_{Q_{X,Y}} - f \otimes g\Big\|^2\right)\\ &= \mathbb{E}_{Q_{X,Y}}\left[f^{\mathrm{T}}(X)g(Y)\right] - \left(\mathbb{E}_{R_X}\left[f(X)\right]\right)^{\mathrm{T}}\mathbb{E}_{R_Y}\left[g(Y)\right] - \frac{1}{2}\cdot\mathrm{tr}\left(\Lambda_f\Lambda_g\right),\end{aligned}$$

with $\Lambda_f = \mathbb{E}_{R_X}\left[f(X)f^{\mathrm{T}}(X)\right]$ and $\Lambda_g = \mathbb{E}_{R_Y}\left[g(Y)g^{\mathrm{T}}(Y)\right]$.

Then, we define the H-score associated with the heterogeneous datasets shown in Table 1, as

$$\mathscr{H}_{\mathrm{m}}(f, g) \triangleq \eta_0 \cdot \mathscr{H}\big(f, g; \hat{P}^{(0)}_{X_1,X_2,Y}\big) + \eta_1 \cdot \mathscr{H}\left(\tau_1(f), g; \hat{P}^{(1)}_{X_1,Y}\right) + \eta_2 \cdot \mathscr{H}\left(\tau_2(f), g; \hat{P}^{(2)}_{X_2,Y}\right), \tag{95}$$

where we have defined conditional expectation operators $\tau_i, i = 1, 2$ as in (82), with respect to the distribution R_{X_1,X_2}. By applying the same nesting configuration C_{BI}, we can obtain the corresponding nested H-score

$$\mathscr{H}_{\mathrm{m}}\left(\begin{bmatrix}\bar{f}\\ f\end{bmatrix}, \begin{bmatrix}\bar{g}\\ g\end{bmatrix}; \mathsf{C}_{\mathsf{BI}}\right) = \mathscr{H}_{\mathrm{m}}(\bar{f}, \bar{g}) + \mathscr{H}_{\mathrm{m}}\left(\begin{bmatrix}\bar{f}\\ f\end{bmatrix}, \begin{bmatrix}\bar{g}\\ g\end{bmatrix}\right). \tag{96}$$

Then, we have the following theorem, which extends Corollary 29 to the heterogeneous datasets. A proof is provide in Appendix C.19.

Theorem 34 *Given* $\bar{k} \geq \mathrm{rank}\Big(\pi_{\mathsf{B}}\Big(\tilde{\ell}_{P^{(\text{est})}_{X_1,X_2,Y}}\Big)\Big)$, *the nested H-score* $\mathscr{H}_{\mathrm{m}}\left(\begin{bmatrix}\bar{f}\\ f\end{bmatrix}, \begin{bmatrix}\bar{g}\\ g\end{bmatrix}; \mathsf{C}_{\mathsf{BI}}\right)$ *as defined in* (96) *is maximized if and only if*

$$\bar{f} \otimes \bar{g} = \pi_{\mathsf{B}}\Big(\tilde{\ell}_{P^{(\text{est})}_{X_1,X_2,Y}}\Big), \qquad f \otimes g = \zeta_{\leq k}\left(\pi_{\mathsf{I}}\Big(\tilde{\ell}_{P^{(\text{est})}_{X_1,X_2,Y}}\Big)\right). \tag{97}$$

We can also use the refined configuration $\mathcal{C}^{\star}_{\mathsf{BI}}$ to obtain modal decomposition of the dependence components. The inference models can be built by assembling learned features, as we have discussed in Section 6.3.

Furthermore, we can show that the estimation $P^{(\mathrm{est})}_{X_1,X_2,Y}$ coincides with the maximum likelihood estimation (MLE) in a local analysis regime. Formally, let $\mathbb{P}\{\mathscr{D}_0, \mathscr{D}_1, \mathscr{D}_2; Q_{X_1,X_2,Y}\}$ denote the probability of observing datasets $\mathscr{D}_0, \mathscr{D}_1, \mathscr{D}_2$, when all data samples are independently generated by $Q_{X_1,X_2,Y}$. Then, we can write the MLE solution as

$$P^{(\mathrm{ML})}_{X_1,X_2,Y} \triangleq \operatorname*{arg\,max}_{Q_{X_1,X_2,Y}} \mathbb{P}\{\mathscr{D}_0, \mathscr{D}_1, \mathscr{D}_2; Q_{X_1,X_2,Y}\}, \tag{98}$$

and we have the following characterization. A proof is provided in Appendix C.20.

Theorem 35 *If $L(R_{X_1,X_2,Y}) = O(\epsilon^2)$, then we have*

$$P^{(\mathrm{ML})}_{X_1,X_2,Y}(x_1,x_2,y) = P^{(\mathrm{est})}_{X_1,X_2,Y}(x_1,x_2,y) + o(\epsilon), \quad \textit{for all } x_1,x_2,y. \tag{99}$$

7. Experimental Verification

To verify the learning algorithms as well as established theoretical properties, we design a series of experiments with various types of data. The main goal is to compare the features learned by neural feature extractors and the corresponding theoretical results. To allow such comparisons, we generate data from given probability distributions of which we know the analytical form of optimal features. The source codes for all experiments are available at `github.com/XiangxiangXu/NFE`, and we defer the implementation details to Appendix D.

7.1 Learning Maximal Correlation Functions

We first consider learning dependence modes, i.e., maximal correlation functions from sample pairs of (X,Y), by maximizing the nested H-score (38). We verify the effectiveness by experiments on both discrete and continuous data, and also discuss one application in analyzing sequential data.

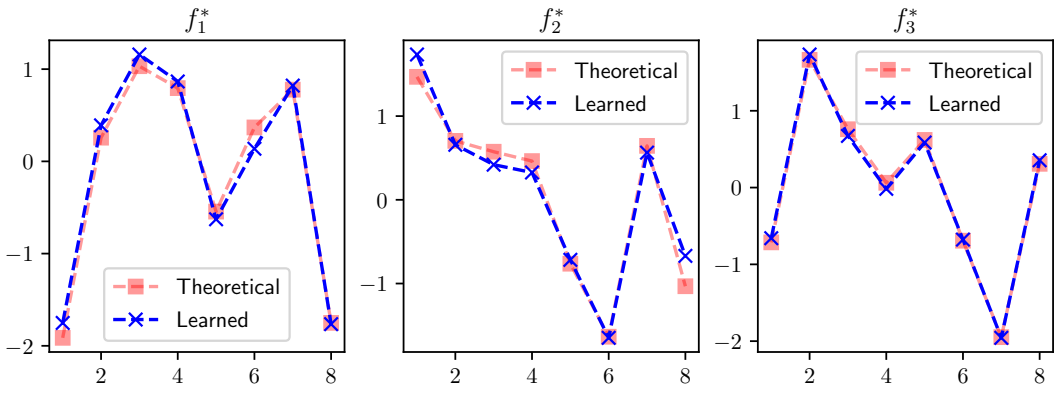
7.1.1 Discrete Data

The simplest case for dependence learning is when X and Y are both discrete with small alphabet sizes $|\mathcal{X}|$ and $|\mathcal{Y}|$. In this case, we can design neural feature extractors with ideal expressive powers. Suppose $\mathcal{X} = \{1,\dots,|\mathcal{X}|\}$, then we can express $f \in \mathcal{F}^k_{\mathcal{X}}$ on $\mathcal{X}$ by first mapping each $i \in \mathcal{X}$ to i-th standard basis vector in $\mathbb{R}^{|\mathcal{X}|}$, also known as the "one-hot encoding" in practice, and then applying a linear function $\mathbb{R}^{|\mathcal{X}|} \to \mathbb{R}^k$ to the mapped result, which we implement by using a linear layer. Then, any $f \in \mathcal{F}^k_{\mathcal{X}}$ can be expressed in this way by setting corresponding weights in the linear layer. Similarly, we can express $g \in \mathcal{F}^k_{\mathcal{Y}}$ using another linear layer.

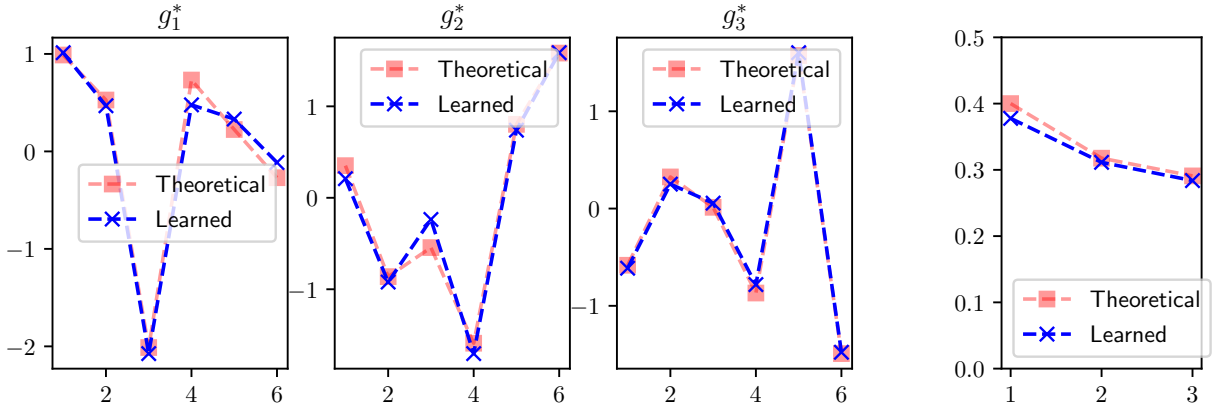
In the experiment, we set $|\mathcal{X}| = 8$, $|\mathcal{Y}| = 6$, and randomly generate a $P_{X,Y} \in \mathcal{P}^{\mathcal{X}\times\mathcal{Y}}$. We generate $N = 30\,000$ training samples from $P_{X,Y}$, and learn $k = 3$ dimensional features f, g by maximizing $\mathscr{H}(f,g;\{(1)^k;\mathcal{F}_{\mathcal{X}};\mathcal{F}_{\mathcal{Y}}\})$. Then, we normalize each f_i, g_i to obtain corresponding estimations of f^*_i, g^*_i, and σ_i by applying (40). We show the estimated features and singular values in Figure 14, which are consistent with the corresponding theoretical values computed from $P_{X,Y}$.

7.1.2 Continuous Data

We proceed to consider a continuous dataset with degenerate dependence modes, i.e., the singular values σ_i's are not all distinct.



(a) $f_i^*(x)$ for each $i = 1, 2, 3$



(b) $g_i^*(y)$ for each $i = 1, 2, 3$

(c) σ_i

Figure 14: Top three dependence modes learned from a discrete dataset, which are consistent with the theoretical results.

In particular, we consider X, Y taking values from $\mathcal{X} = \mathcal{Y} = [-1, 1]$, where the joint probability density function $p_{X,Y}$ takes a raised cosine form:

$$p_{X,Y}(x, y) = \frac{1}{4} \cdot [1 + \cos(\pi(x - y))], \quad (x, y) \in [-1, 1]^2. \tag{100}$$

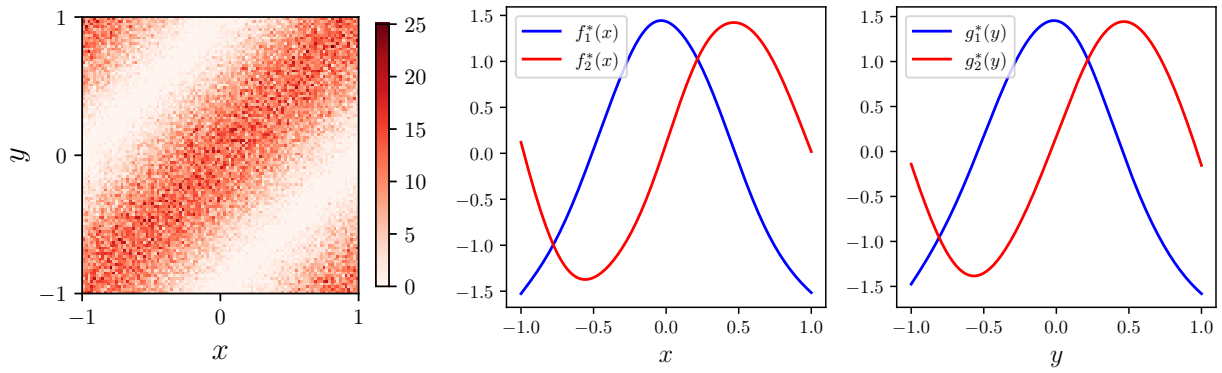
Then, it can be verified that the corresponding marginal distributions of X, Y are uniform distributions $p_X = p_Y = \mathrm{Unif}([-1, 1])$. In addition, the resulting CDK function is

$$\mathfrak{i}_{X;Y}(x, y) = \frac{p_{X,Y}(x, y) - p_X(x)p_Y(y)}{p_X(x)p_Y(y)} = \cos(\pi(x - y)). \tag{101}$$

Note that we have $\cos(\pi(x - y)) = \cos(\pi x + \theta_0) \cdot \cos(\pi y + \theta_0) + \sin(\pi x + \theta_0) \cdot \sin(\pi y + \theta_0)$, for any $\theta_0 \in [-\pi, \pi)$. Therefore, we have $\mathrm{rank}(\mathfrak{i}_{X;Y}) = 2$, and the associated dependence modes are given by $\sigma_1 = \sigma_2 = 1/2$ and the maximal correlation functions

$$f_1^*(x) = \sqrt{2} \cdot \cos(\pi x + \theta_0), \qquad f_2^*(x) = \sqrt{2} \cdot \sin(\pi x + \theta_0), \tag{102a}$$

$$g_1^*(y) = \sqrt{2} \cdot \cos(\pi y + \theta_0), \qquad g_2^*(y) = \sqrt{2} \cdot \sin(\pi y + \theta_0), \tag{102b}$$



(a) Histogram of Training Data

(b) Learned Dependence Modes

Figure 15: Dependence modes learned from continuous data, in consistent with theoretical results.

for any $\theta_0 \in [-\pi, \pi)$.

During this experiment, we first generate $N = 50\,000$ sample pairs of X, Y for training, with histogram showing in Figure 15a. Then, we learn $k = 2$ dimensional features f_1, f_2 of $\mathcal{X}$ and g_1, g_2 of $\mathcal{Y}$ by maximizing the nested H-score (38), where f and g are parameterized neural feature extractors detailed in Appendix D.1.2. Figure 15b shows the learned functions after the normalization (40). The learned results well match the theoretical results (102): (i) The learned f_1^* and f_2^* are sinusoids differ in phase by $\pi/2$, and (ii) g_i^* coincides with f_i^*, for each $i = 1, 2$. It is also worth mentioning that due to the degeneracy $\sigma_1 = \sigma_2$, the initial phase θ_0 in learned sinusoids (102) can be different during each run of the training algorithm.

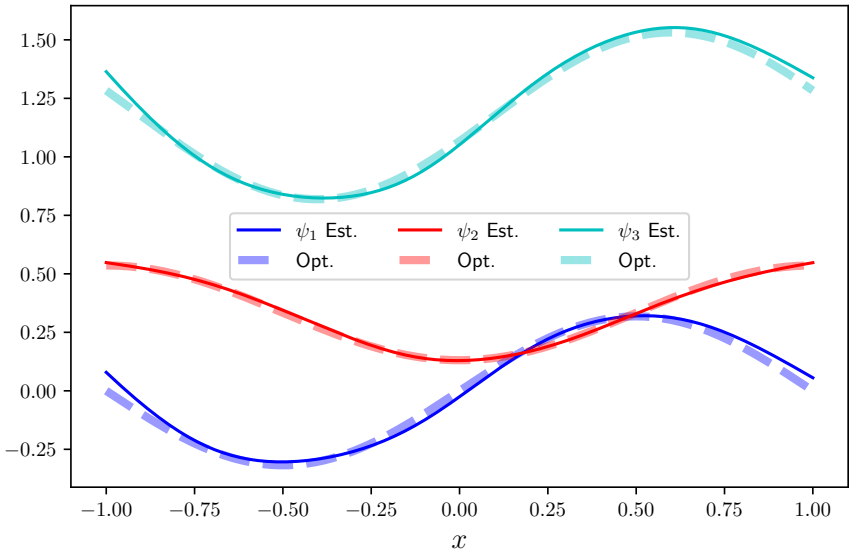


Figure 16: MMSE estimators $\mathbb{E}[\psi_i(Y)|X = x]$ obtained from learning dependence modes, in comparison with theoretical results.

Based on the learned dependence modes, we then demonstrate estimating functions of Y based on observed $X = x$. Here, we consider the functions $\psi_1(y) = y, \psi_2(y) = y^2, \psi_3(y) = e^y$. From Proposition 12, we can compute the learned MMSE estimator $\mathbb{E}[\psi_i(Y)|X = x]$ for each i, by estimating $\mathbb{E}[\psi_i(Y)]$ and $\Lambda_{\psi_i,g} = \mathbb{E}\left[\psi_i(Y)g^{\mathrm{T}}(Y)\right]$ from the training set and then applying (26).

For comparison, we compute the theoretical values

$$\mathbb{E}[\psi_i(Y)|X = x] = \int_{-1}^{1} p_{Y|X}(y|x)\psi_i(y)\,dy,$$

with $p_{Y|X}(y|x) = \frac{1}{2} \cdot [1 + \cos(\pi(y - x))]$, which gives

$$\mathbb{E}[Y|X = x] = \frac{1}{\pi} \cdot \sin(\pi x), \qquad \mathbb{E}\left[Y^2\middle|X = x\right] = \frac{1}{3} - \frac{2}{\pi^2}\cos(\pi x), \tag{103a}$$

$$\mathbb{E}\left[e^Y\middle|X = x\right] = \frac{e^2 - 1}{2e(1 + \pi^2)} \cdot (\pi \sin(\pi x) - \cos(\pi x) + \pi^2 + 1). \tag{103b}$$

Figure 7.1.2 shows the estimators obtained by applying (26) on the learned features, which are consistent with the theoretically optimal estimators given by (103).

7.1.3 Sequential Data

We proceed with an example of learning dependence modes among sequence pairs. For simplicity, we consider binary sequences $\underline{X}$ and $\underline{Y}$, of lengths l and m, respectively. Suppose we have the Markov relation $\underline{X} - U - V - \underline{Y}$ for some unobserved binary factors $U, V \in \mathcal{U} = \mathcal{V} = \{0, 1\}$. In addition, we assume[14] $\underline{X} = (X_1, \ldots, X_l)^{\mathrm{T}}, \underline{Y} = (Y_1, \ldots, Y_m)^{\mathrm{T}}$ satisfy

$$\underline{X}|U = i \sim \mathrm{BMS}(l, q_i), \quad \underline{Y}|V = i \sim \mathrm{BMS}(m, q_i), \quad \text{for } i = 0, 1, \tag{104}$$

where $\mathrm{BMS}(l, q)$ denotes the distribution of a binary first-order Markov sequence of length l with state flipping probability q. The corresponding state transition diagram is shown in Figure 17a. Therefore, if $\underline{Z} \sim \mathrm{BMS}(l, q)$, then $Z_1 \sim \mathrm{Unif}(\{0, 1\})$ and $(Z_1, \ldots, Z_l)$ forms a first order Markov chain over binary states $\{0, 1\}$, with flipping probability $\mathbb{P}\{Z_{i+1} \neq Z_i | Z_i = z\} = q$ for both $z = 0, 1$. Formally, $\underline{Z} \sim \mathrm{BMS}(l, q)$ if and only if

$$P_{\underline{Z}}(\underline{z}) = \frac{1}{2} \cdot \prod_{i=1}^{l-1} \left[(1 - q)^{\delta_{z_i z_{i+1}}} \cdot q^{1 - \delta_{z_i z_{i+1}}}\right] \quad \text{for all } (z_1, \ldots, z_l) \in \{0, 1\}^l.$$

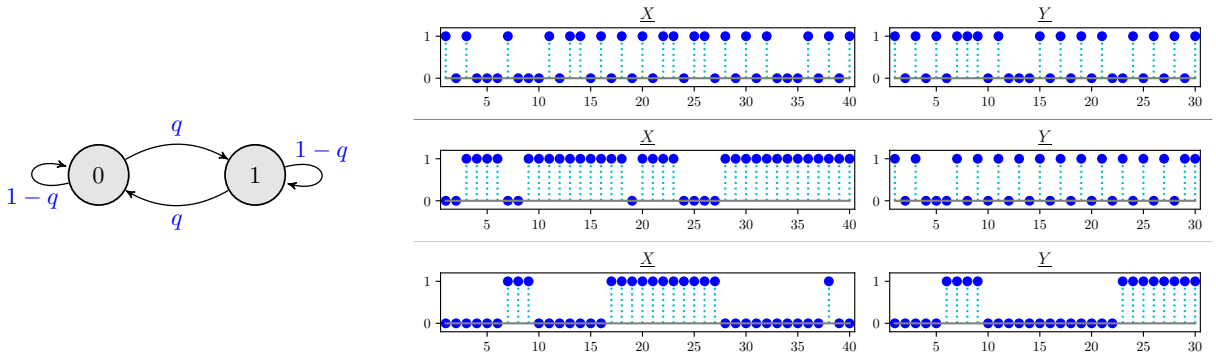
As a consequence, the resulting alphabets are $\mathcal{X} = \{0, 1\}^l, \mathcal{Y} = \{0, 1\}^m$, with sizes $|\mathcal{X}| = 2^l, |\mathcal{Y}| = 2^m$. In our experiment, we set $l = 40, m = 30, q_0 = 0.1, q_1 = 0.9$, and use the following joint distribution $P_{U,V}$:

Prob.	$U = 0$	$U = 1$
$V = 0$	0.1	0.2
$V = 1$	0.4	0.3

We generate $N = 50\,000$ training sample pairs of $\underline{X}, \underline{Y}$, with instances shown in Figure 17b. We also generate $N' = 10\,000$ sample pairs in the same manner, as the testing dataset.

Then, we learn $k = 1$ dimensional features f and g by maximizing $\mathscr{H}(f, g)$ over the training set. We plot the extracted features in Figure 18. In particular, each point represents an $(f(\underline{x}), g(\underline{y}))$ pair

14. For convenience, we adopt the vector notation to represent sequences.



(a) State Transitions of BMS(l, q) (b) Generated Samples of $\underline{X}, \underline{Y}$ Pair

Figure 17: Sequential Data: Model and Generated $\underline{X}, \underline{Y}$ Samples Pairs

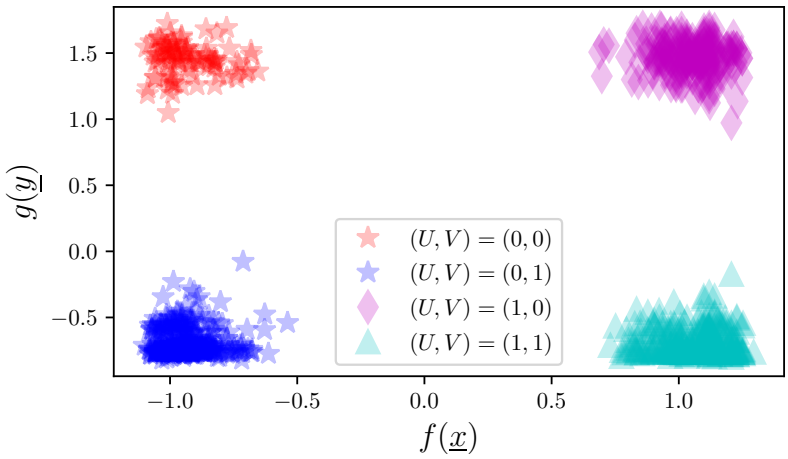


Figure 18: Features Learned from Sequential Data $\underline{X}$ and $\underline{Y}$

evaluated on an instance from testing set, with corresponding values of binary factors (U, V) shown for comparison. For the ease of demonstration, here we plot only $1,000$ sample pairs randomly chosen from the testing set. From the figure, the learned features are clustered according to the underlying factors. This essentially reconstructs the hidden factors U, V. For example, one can apply a standard clustering algorithm on the features, e.g., k-means (Hastie et al., 2009), then count the proportion of each cluster, to obtain an estimation of $P_{U,V}$ up to permutation of symbols.

For a closer inspection, we can compare the learned features with the theoretical results, formalized as follows. A proof is provided in Appendix C.21.

Proposition 36 *Suppose $\underline{X}, \underline{Y}$ satisfy the Markov relation $\underline{X} - U - V - \underline{Y}$ with $U, V \in \{0, 1\}$ and the conditional distributions* (104)*. Then, we have* $\operatorname{rank}(\mathfrak{i}_{X;Y}) \leq 1$*, and the corresponding maximal correlation functions f_1^*, g_1^* satisfy*

$$f_1^*(\underline{x}) = c \cdot [\tanh(2w \cdot \varphi(\underline{x}) + b_U) - \tanh(b_U)], \tag{105a}$$

$$g_1^*(\underline{y}) = c' \cdot \left[\tanh\left(2w \cdot \varphi(\underline{y}) + b_V\right) - \tanh(b_V)\right], \tag{105b}$$

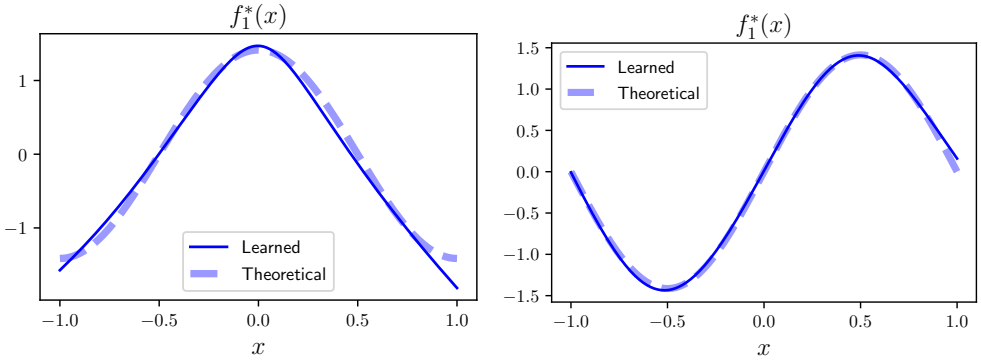
for some $c, c' \in \mathbb{R}$*, where* $w \triangleq \frac{1}{2}\log\frac{q_1}{q_0}, b_U \triangleq \frac{1}{2}\log\frac{P_U(1)}{P_U(0)}, b_V \triangleq \frac{1}{2}\log\frac{P_V(1)}{P_V(0)}$*, and where we have defined* $\varphi\colon \{0,1\}^* \to \mathbb{R}$*, such that for each* $\underline{z} = (z_1, \ldots, z_l)^{\mathrm{T}} \in \{0,1\}^l$*, we have* $\varphi(\underline{z}) \triangleq \dfrac{l-1}{2} - \displaystyle\sum_{i=1}^{l-1} \delta_{z_i z_{i+1}}$.

Then, we compute the correlation coefficients between $f(\underline{X})$ and $f_1^*(\underline{X})$, and between $g(\underline{Y})$, $g_1^*(\underline{Y})$, respectively, using sample pairs in the testing set. The absolute values of both correlation coefficients are greater than 0.99, demonstrating the effectiveness of the learning algorithm.

7.2 Learning With Orthogonality Constraints

We verify the feature learning with orthogonality constraints presented in Section 4.4 on the same dataset used for Section 7.1.2. Here, we consider the settings $\bar{k} = k = 1$, i.e., we learn one-dimensional feature $f(x)$ uncorrelated to given one dimensional feature $\phi \in \mathcal{F}_{\mathcal{X}}$.

Note that without any orthogonality constraints [cf. (102)], the optimal feature will be sinusoids with any initial phase, i.e., $f_1^*(x) = \sqrt{2} \cdot \cos(\pi x + \theta_0)$ for any $\theta_0 \in [-\pi, \pi)$. Here, we consider the following two choices of ϕ, $(x \mapsto x)$ and $(x \mapsto x^2)$, which are even and odd functions, respectively. Since the underlying p_X is uniform on $[-1, 1]$, we can verify the optimal features under the two constraints are $f_1^*(x) = \sqrt{2}\cos(\pi x)$ for $\phi(x) = x$, and $f_1^*(x) = \sqrt{2}\sin(\pi x)$ for $\phi(x) = x^2$.



(a) $\phi(x) = x$ (b) $\phi(x) = x^2$

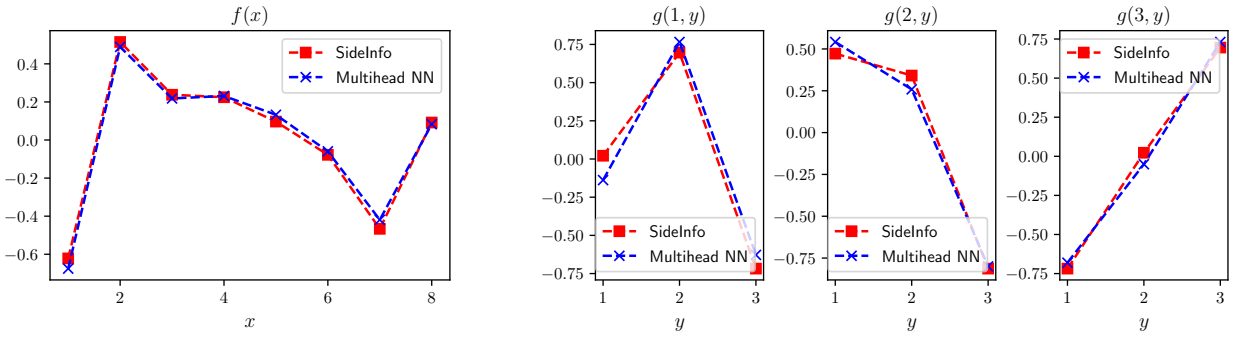
Figure 19: Learning features uncorrelated to given ϕ under different settings of ϕ. The learned results are compared with theoretical results.

By maximizing the nested H-score restricted to $\bar{f} = \phi$ [cf. (52)], we can learn the optimal feature f_1^*, as shown in Figure 19. The learned features are in consistent with the theoretical ones, validating the effectiveness of the learning algorithm.

7.3 Learning With Side Information

We design an experiment to verify the connection between our learning algorithm and the multitask classification DNN, as demonstrated in Theorem 27.

In particular, we consider the discrete X, Y, S with $|\mathcal{X}| = 8, |\mathcal{S}| = |\mathcal{Y}| = 3$, and randomly choose a joint distribution $P_{X,S,Y}$ on $\mathcal{X} \times \mathcal{S} \times \mathcal{Y}$. Then, we generate $N = 50\,000$ training samples of (X, S, Y) triples. In our implementation, we set $\bar{k} = |\mathcal{S}| - 1 = 2, k = 1$ and maximize the nested H-score configured by $\mathcal{C}_{\mathsf{MC}}$ [cf. (59)] on the training set.



(a) Learned feature $f(x)$

(b) Learned feature $g(s, y)$

Figure 20: Experimental verification of the connection between learning with side information and training a multihead neural network.

For comparison, we also train the multihead network shown in Figure 11, where we maximize the log-likelihood function (70) to learn the corresponding feature f and weight matrices G_s for all $s \in \mathcal{S}$. Then, we convert the weights to $g \in \mathcal{F}_{\mathcal{S}\times\mathcal{Y}}$ via the correspondence [cf. (69)] $g(s, y) = G_s(1, y)$. The features learned by our algorithm (labeled as "SideInfo") and the multihead neural network are shown in Figure 20, where the results are consistent.

7.4 Multimodal Learning With Missing Modalities

To verify the multimodal learning algorithms presented in Section 6, we consider multimodal classification problems in two different settings. Suppose X_1, X_2 are multimodal data variables, and $Y \in \mathcal{Y} = \{-1, 1\}$ denotes the binary label to predict. In the first setting, we consider the training set with complete (X_1, X_2, Y) samples. In the second setting, only the pairwise observations of (X_1, X_2), (X_1, Y), and (X_2, Y) are available, presented in three different datasets.

In both settings, we set $\mathcal{X}_1 = \mathcal{X}_2 = [-1, 1]$ with

$$P_{X_1,X_2}(x_1, x_2) = \frac{1}{4} \cdot [1 + \cos(2\pi(x_1 - x_2))]\,. \tag{106}$$

We consider predicting Y based on the learned features, where some modality in X_1, X_2 might be missing during the prediction.

7.4.1 Learning from Complete Observations

We consider the X_1, X_2, Y dependence specified by (106) and the conditional distribution

$$P_{Y|X_1,X_2}(y|x_1, x_2) = \frac{1}{2} + \frac{y}{4} \cdot [\cos(\pi x_1) + \cos(\pi x_2) + \cos(\pi(x_1 + x_2))] \tag{107}$$

for $x_1, x_2 \in [-1, 1]$ and $y = \pm 1$. It can be verified that P_Y satisfies $P_Y(1) = P_Y(-1) = \frac{1}{2}$. The corresponding CDK function and dependence components [cf. (74)] are given by

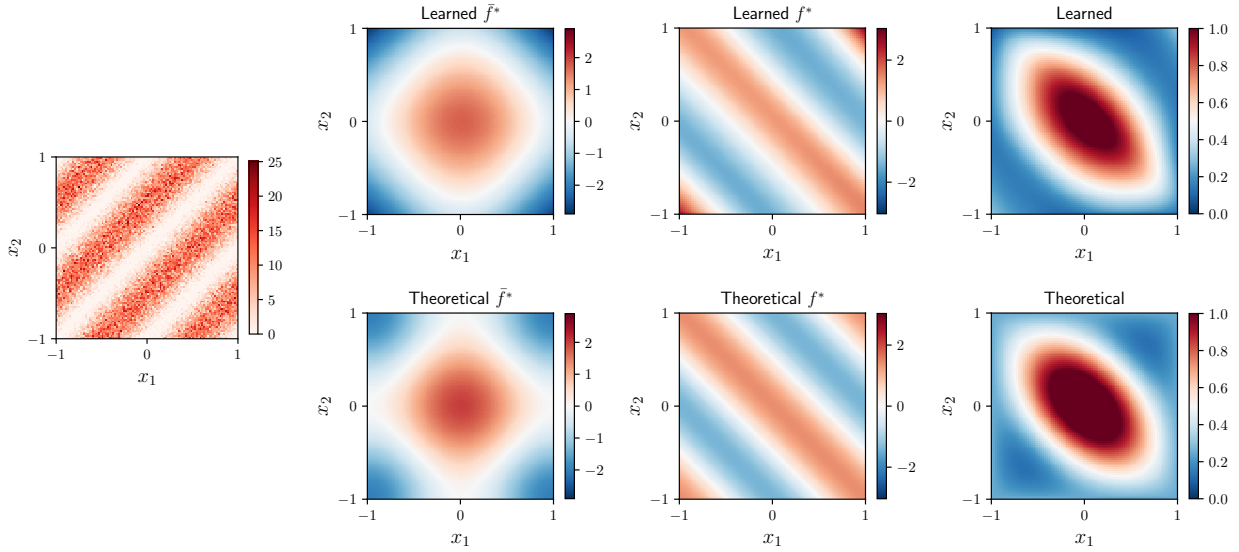
$$\mathsf{i}_{X_1,X_2;Y}(x_1, x_2, y) = \frac{y}{2} \cdot [\cos(\pi x_1) + \cos(\pi x_2) + \cos(\pi(x_1 + x_2))] \tag{108a}$$

$$[\pi_{\mathsf{B}}(\mathsf{i}_{X_1,X_2;Y})](x_1, x_2, y) = \frac{y}{2} \cdot [\cos(\pi x_1) + \cos(\pi x_2)]\,, \tag{108b}$$

$$[\pi_{\mathsf{I}}(\mathfrak{i}_{X_1,X_2;Y})](x_1,x_2,y) = \frac{y}{2}\cdot\cos(\pi(x_1+x_2)). \tag{108c}$$

Therefore, we have $\operatorname{rank}(\pi_{\mathsf{B}}(\mathfrak{i}_{X_1,X_2;Y})) = \operatorname{rank}(\pi_{\mathsf{I}}(\mathfrak{i}_{X_1,X_2;Y})) = 1$, and the functions obtained from modal decompositions [cf. (91)] are $\bar{g}_1^*(y) = g_1^*(y) = y$ and

$$\bar{f}_1^*(x_1,x_2) = \cos(\pi x_1) + \cos(\pi x_2), \qquad f_1^*(x_1,x_2) = \sqrt{2}\cdot\cos(\pi(x_1+x_2)). \tag{109}$$



(a) X_1, X_2 Histogram (b) $\bar{f}_1^*(x_1,x_2)$ (c) $f_1^*(x_1,x_2)$ (d) $P_{Y|X_1,X_2}(1|x_1,x_2)$

Figure 21: Features and posterior probability learned from multimodal data X_1, X_2, Y, in comparison with theoretical results.

In the experiment, we first generate $N = 50\,000$ triples of (X_1, X_2, Y) for training. The histogram of (X_1, X_2) pair is shown in Figure 21a. Then, we set $\bar{k} = k = 1$, and learn the features $\bar{f}, f, \bar{g}, g$ by maximizing the nested H-score configured by $\mathcal{C}_{\mathsf{BI}}$ [cf. (78)]. We then normalize $\bar{f}$, f to obtain the estimated $\bar{f}_1^*$ and f_1^*, and compute the posterior probability $P_{Y|X_1,X_2}(1|x_1,x_2)$ based on (83a). The results of learned features $\bar{f}_1^*$, f_1^* and posterior $P_{Y|X_1,X_2}(1|x_1,x_2)$ are shown in Figure 21, which are consistent with the theoretical results.

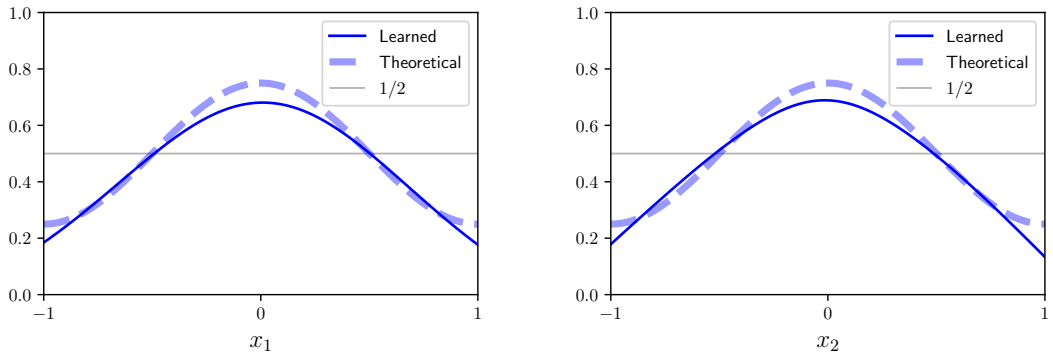
We then consider the prediction problem with missing modality, i.e., predict label Y based on unimodal data X_1 or X_2. In particular, based on learned $\bar{f} = \bar{f}^{(1)} + \bar{f}^{(2)}$, we train two separate networks to operate as τ_1 and τ_2, then apply (83b) and (83c) to estimate the posteriors $P_{Y|X_1}$ and $P_{Y|X_2}$. Then, for each $i = 1, 2$, the MAP prediction of Y based on observed $X_i = x_i$ can be obtained by comparing $P_{Y|X_i}(1|x_i)$ with the threshold $1/2$, via

$$\arg\max_{y\in\mathcal{Y}} P_{Y|X_i}(y|x_i) = \begin{cases} 1 & \text{if } P_{Y|X_i}(1|x_i) > 1/2, \\ -1 & \text{if } P_{Y|X_i}(1|x_i) \le 1/2. \end{cases}$$

We plot the estimated results in Figure 22, in comparison with the threshold $1/2$ and the theoretical values

$$P_{Y|X_i}(y|x) = \frac{1}{2} + \frac{y}{4}\cdot\cos(\pi x), \quad \text{for } i = 1, 2. \tag{110}$$

From the figure, the estimated posteriors $P_{Y|X_1}, P_{Y|X_2}$ have consistent trends with the ground truth posteriors, and the induced Y predictions are well aligned.



(a) $P_{Y|X_1}(1|x_1)$ (b) $P_{Y|X_2}(1|x_2)$

Figure 22: Label prediction from unimodal data using learned features

7.4.2 Learning from Pairwise Observations

We proceed to consider the multimodal learning with only pairwise observations. Specifically, we consider the joint distribution of (X_1, X_2, Y), specified by (106) and

$$P_{Y|X_1,X_2}(y|x_1,x_2) = \frac{1}{2} + \frac{y}{4} \cdot [\cos(\pi x_1) + \cos(\pi x_2)]\,. \tag{111}$$

for $x_1, x_2 \in [-1, 1]$ and $y = \pm 1$. It can be verified that $P_Y(1) = P_Y(-1) = \frac{1}{2}$, and the associated CDK function satisfies

$$\mathfrak{i}_{X_1,X_2;Y}(x_1,x_2,y) = \frac{y}{2} \cdot [\cos(\pi x_1) + \cos(\pi x_2)]\,. \tag{112}$$

and $\mathfrak{i}_{X_1,X_2;Y} = \pi_{\mathsf{B}}(\mathfrak{i}_{X_1,X_2;Y})$. Therefore, the interaction dependence component $\pi_{\mathsf{I}}(\mathfrak{i}_{X_1,X_2;Y}) = 0$, and the joint dependence can be learned from all pairwise samples, as we discussed in Section 6.5.1.

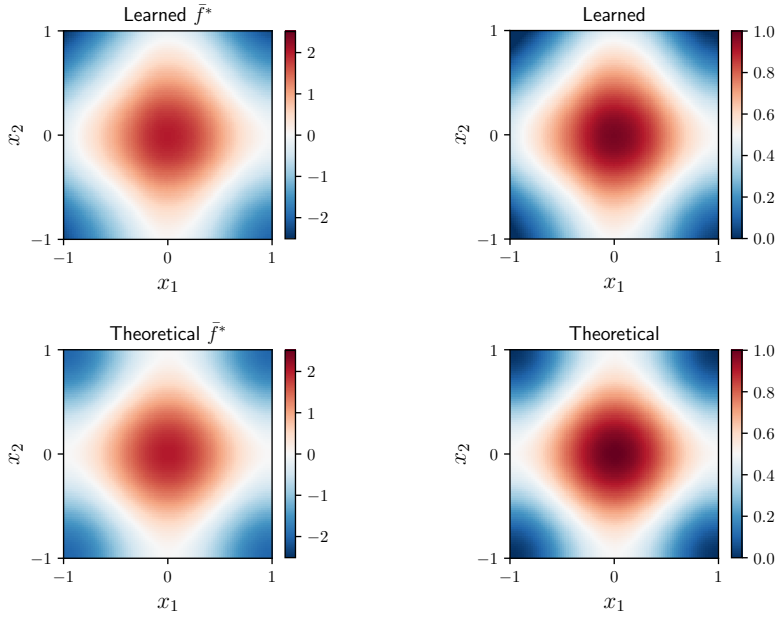
We then construct an experiment to verify learning joint dependence from all pairwise observations. Specifically, we generate $N = 50\,000$ triples of (X_1, X_2, Y) from (106) and (111). Then, we adopt the decomposition (89) on each triple, to obtain three pairwise datasets with samples of (X_1, X_2), (X_1, Y), (X_2, Y), where each dataset has N sample pairs.

We use these three pairwise datasets for training and learn one dimensional $\bar{f} \in \mathcal{F}_{\mathcal{X}}, \bar{g} \in \mathcal{F}_{\mathcal{Y}}$ that maximize $\mathscr{H}(\bar{f}, \bar{g})$. Here, we compute $\mathscr{H}(\bar{f}, \bar{g})$ based on the minibatches from the three pairwise datasets, according to (93). Based on learned $\bar{f}, \bar{g}$, we then compute the normalized $\bar{f}_1^*$ and posterior distribution $P_{Y|X_1,X_2}(1|x_1,x_2)$, as shown in Figure 23, where the learned results match the theoretical values.

Similar to the previous setting, we consider the unimodal prediction problem, and show the estimated results in Figure 24. It is worth noting that from (108b), (112), the joint distributions $P_{X_1,X_2,Y}$ in both settings contain the same bivariate dependence component. Therefore, the theoretical results for $\bar{f}_1^*$ and $P_{Y|X_1}, P_{Y|X_2}$ are the same.

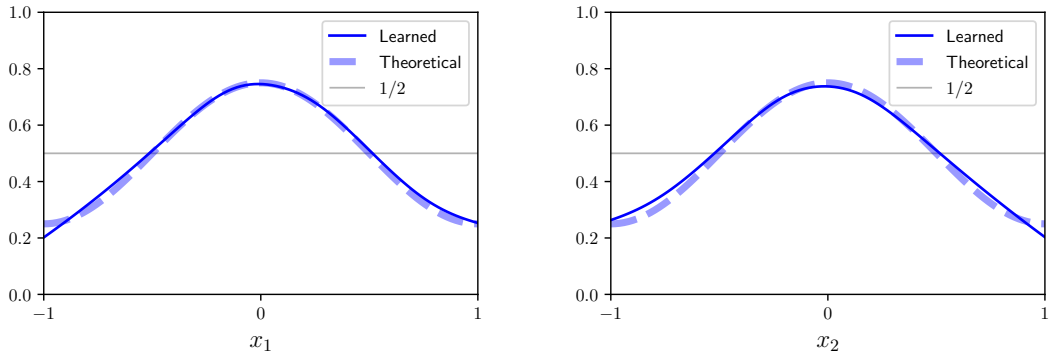
8. Related Works

Maximal Correlation Functions: Optimality, Learning Algorithms, and Applications The Hirschfeld–Gebelein–Rényi (HGR) maximal correlation (Hirschfeld, 1935; Gebelein, 1941; Rényi, 1959) provides an important connection between statistical dependence and function space. The same concept has been studied with various formulations, and often in different terminologies,



(a) $\bar{f}_1^*(x_1, x_2)$ (b) $P_{Y|X_1,X_2}(1|x_1, x_2)$

Figure 23: Features and posterior probability learned from pairwise datasets of $(X_1, X_2), (X_1, Y)$, and (X_2, Y), in comparison with theoretical results.



(a) $P_{Y|X_1}(1|x_1)$ (b) $P_{Y|X_2}(1|x_2)$

Figure 24: Label prediction from unimodal data, with features learned from pairwise datasets.

such as correspondence analysis (Greenacre, 2017), functional canonical variates (Buja, 1990), and principal inertial components (du Pin Calmon et al., 2017). See, e.g., (Huang et al., 2024, Section II), for a detailed discussion. The optimality of HGR maximal correlation functions has been extensively studied in literature, e.g., as the optimal transformations in regression problems (Breiman and Friedman, 1985; Buja, 1990). In particular, if one variable is categorical and represents the class label, the maximal correlation functions also provide the best inter-class separability (Xu and Huang, 2020, Section III-B). The maximal correlation functions also play a fundamental role in the discussion of informative feature extraction and local information geometry, as we will detail next. The first practical algorithm for learning such functions is the alternating conditional expectations

(ACE) algorithm (Breiman and Friedman, 1985; Buja, 1990), which learns the top dependence mode and is mostly used for processing low-dimensional data. To tackle high-dimensional data, recent developments focused on learning maximal correlation functions by DNNs. Specifically, the H-score was derived from the low-rank approximation of the canonical dependence kernel, referred as the Soft-HGR objective (Wang et al., 2019). It was also introduced as the local approximation of the log loss function (Xu et al., 2022), as we have discussed in Section 3.4. In an independent line of work (HaoChen et al., 2021), a special form of H-score was proposed for self-supervised learning tasks, referred to as the *spectral contrastive loss* therein. There are also other formulations and algorithms proposed for computing the maximal correlation functions; see, e.g., Andrew et al. (2013) and Hsu et al. (2021). Compared to the low-rank approximation formulation, such approaches typically require explicit whitening procedures or computations of matrix inverses, which are less applicable for learning high-dimensional features. The sample complexity for learning maximal correlation functions have been investigated recently; see, e.g., Huang and Xu (2020); Makur et al. (2020). The maximal correlation learning, particularly the H-score maximization formulation, has been widely applied in feature extraction for multimodal data, e.g., Wang et al. (2019) and the follow-up works. For example, Xu and Huang (2020) developed maximal correlation regression (MCR), which uses maximal correlation functions to solve classification tasks.

Informative Feature Extraction and Local Information Geometry Huang et al. (2024) has provided an in-depth characterization of informative features by applying information-theoretic tools, with a focus on bivariate learning problems and the local analysis regime. Under such settings, it was shown that a series of statistical and learning formulations lead to the same optimal features, characterized as the HGR maximal correlation functions. Examples of these formulations include the information bottleneck (Tishby et al., 2000) and a cooperative game. There are similar information-theoretic discussions on multivariate problems. For example, Xu and Zheng (2022) presented the information-theoretic optimality of the features defined in (66). The features introduced in (91a) were also studied by Xu and Huang (2021) in characterizing distributed hypothesis testing problems.

Decomposition of Probability Distributions The decomposition of probability distributions has been studied, particularly in the context of information geometry (Amari and Nagaoka, 2000). For example, Amari (2001) established a decomposition in distribution space and also investigated the maximum entropy formulation (87), cf. (Amari, 2001, Theorem 7). The information geometry induced by classification DNNs, with softmax activation and log loss function, was investigated in Xu et al. (2018), which corresponds to the optimal features without the weak dependence assumption [cf. Proposition 14].

The learning framework we presented is built upon these existing works and the perspectives from information theory, statistical analysis, and machine learning. The key component is a unified function-space view of feature learning design, which has provided nontrivial extensions to existing works. In particular, with the function-space viewpoint and nesting technique, we have developed several operations on general multivariate statistical dependence, with the existing bivariate learning algorithm (Wang et al., 2019; Xu and Huang, 2020) being atomic operations and special cases. Our developments have also revealed the learning implications of existing theoretical developments, e.g., Amari and Nagaoka (2000) and Huang et al. (2024).

9. Conclusions and Discussion

We have presented a framework for designing learning systems with neural feature extractors, which allows us to learn informative features and assemble them to build different inference models. Based on the feature geometry, we use feature representations as the interface, and convert learning problems to corresponding function-space operations. We then introduce the nesting technique for implementing such operations, which provides a systematic design of both feature learning and feature assembling modules. We demonstrate its applications in learning multivariate dependence by considering conditional inference and multimodal learning problems.

The developed framework has provided a unified solution to general feature-centric learning problems, where we have used DNNs as building blocks to directly represent the statistical dependence. The connection between feature and statistical dependence enables principled learning algorithm designs, especially in tackling complicated multivariate dependence. Such designs also provide a learning-based computational approach for investigating the statistical dependence behind high-dimensional data with often complicated structures.

9.1 Applications and Extensions

For simplicity, our presentation focused on theoretical concepts and basic settings, which can be readily applied to practical learning scenarios and extended to more complicated settings. For instance, we can establish the feature optimality in multivariate problems by extending the bivariate characterizations (Huang et al., 2024; Xu and Huang, 2020); see, e.g, the discussions in Xu and Zheng (2022).

For applications, the results in Proposition 12 can be applied to classification and estimation tasks, e.g., image classification (Xu and Huang, 2020). Similarly, we can employ the results in Section 4.4 to address physical or engineering constraints induced by practical learning applications. We can also apply the learning framework to process multimodal data dependencies and learn useful features, as discussed in Section 5 and Section 6. In all such examples, we can incorporate pretrained models to reduce data requirements and computational costs.

Our discussions on the dependence decomposition framework can also be extended to general settings. In particular, we can get refined dependence components by iteratively applying the decomposition (43). One example is to decompose the dependence of a random process into different dependence components, with each component representing the dependence at a certain time delay, as discussed in Xu and Zheng (2023b). More generally, the nesting technique can be extended by considering a configuration (cf. Definition 16) with subspace sequences of both $\mathcal{F}_{\mathcal{X}}$ and $\mathcal{F}_{\mathcal{Y}}$.

We can also extend our discussions and analyses on supervised learning examples to other scenarios, e.g., contrastive (Chen et al., 2020) and self-supervised (HaoChen et al., 2021) learning problems. In addition to the neural feature extractors, the feature geometry can also be applied to implicit features, e.g., the feature subspaces in kernel methods. In particular, Xu and Zheng (2023a) developed a quantitative measure of kernel choices and demonstrated a connection between kernel methods and existing feature learning approaches.

9.2 Future Directions

The established framework provides abundant opportunities for further explorations in both theoretical analyses and applications. To highlight the general framework, our development has adopted simplified or idealized assumptions on learning factors, e.g., the network expressive power. By relaxing such idealized assumptions, further investigations can give a better theoretical understanding and also provide practical guidance. Preliminary examples include our discussions on the expressive

power of feature extractors (cf. Section 3.3) and the sample size of training data (cf. Section 6.5.2). More in-depth characterizations on the learning behaviors, e.g., generalizability, can be built upon the spectrum decomposition nature of this learning framework, by using existing analyses on linear operators (Dunford and Schwartz, 1988) or the spectral methods for matrices (Chen et al., 2021). Another interesting direction is to investigate the interaction between feature geometry, the manifold structure of the neural feature extractor, and the geometry of data spaces (Bronstein et al., 2017, 2021). Moreover, the nested H-score and the induced optimization behaviors can be of independent interest to optimization studies, e.g., the landscape and convergence analyses. In terms of applications, the established framework can be extended to other operations on statistical dependence beyond decomposing or learning given dependence. For instance, a potential application is to generate data satisfying certain dependence constraints, e.g., independence or conditional independence, by employing generative models. The integration with generative models, as well as the learning algorithm design, is of practical interest for further studies.

Acknowledgments

This work was supported by the Office of Naval Research (ONR) under grant N00014-19-1-2621.

Appendix A. Data Alphabets in Feature Geometry

We discuss the feature geometry on general data alphabets, e.g., continuous alphabets in Appendix A.1. In Appendix A.2, we briefly summarize the vector and matrix notation conventions established for discrete alphabets, which have been extensively used in related works, e.g., Huang et al. (2024); Xu et al. (2022).

A.1 General Alphabets: Regularity Conditions and Examples

Our development on feature spaces can be extended to general Hilbert spaces (Young, 1988; Weidmann, 2012), where the alphabet $\mathcal{Z}$ and the metric distribution P_Z (cf. Section 2.1.2) are extended to a measurable set and the measure (Weidmann, 2012, Example 6), respectively.

One particularly important example is $\mathcal{Z} = \mathbb{R}^d$ with $d \geq 1$. We consider the random variable $\underline{Z} \in \mathcal{Z}$. For simplicity, we restrict to the case where $\underline{Z}$ has the probability density function $p_{\underline{Z}}$. Then, we define the feature space $\mathcal{F}_{\mathcal{Z}}$ and the inner product on $\mathcal{F}_{\mathcal{Z}}$ as [cf. Section 2.1.2]

$$\mathcal{F}_{\mathcal{Z}} \triangleq \left\{ f \colon \mathcal{Z} \to \mathbb{R}, \int f^2(\underline{z}) p_{\underline{Z}}(\underline{z}) d\underline{z} < \infty \right\}, \quad \text{and} \quad \langle f_1, f_2 \rangle \triangleq \int f_1(\underline{z}) f_2(\underline{z}) p_{\underline{Z}}(\underline{z}) d\underline{z} \text{ for } f_1, f_2 \in \mathcal{F}_{\mathcal{Z}},$$

respectively.

We can define the joint function similarly for $\mathcal{X}$, $\mathcal{Y}$ under the corresponding regularity conditions. For instance, we consider (X, Y) with alphabets $\mathcal{X} = \mathcal{Y} = \mathbb{R}$ and the joint probability density function $p_{X,Y}$. Then, the corresponding definition of the CDK function becomes [cf. (4)]

$$\mathfrak{i}_{X;Y}(x, y) = \frac{p_{X,Y}(x, y)}{p_X(x) p_Y(y)} - 1. \tag{113}$$

Note that to have $\mathfrak{i}_{X;Y} \in \mathcal{F}_{\mathcal{X} \times \mathcal{Y}}$, we need to assume

$$\|\mathfrak{i}_{X;Y}\|^2 = \iint \frac{[p_{X,Y}(x, y)]^2}{p_X(x) p_Y(y)} \, dx dy - 1 < \infty, \tag{114}$$

where the norm is defined with respect to the metric distribution $p_X p_Y$.

Remark 37 *The results can be further generalized to the case where density functions do not necessarily exist. See, e.g., Lancaster (1958) and (Buja, 1990, Proposition 3.1) and for detailed discussions.*

In particular, we have the following result for bivariate normal variables, which has been extensively discussed in literature; see, e.g., Lancaster (1958) and Huang et al. (2024).

Example 2 *For bivariate normal variables* $\begin{bmatrix} X \\ Y \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix} \right)$ *with* $|\rho| < 1$*, we have*

$$\mathfrak{i}_{X;Y}(x, y) = \sum_{i=1}^{\infty} \frac{\rho^i}{i!} \cdot \mathtt{He}_i(x) \cdot \mathtt{He}_i(y). \tag{115}$$

where for each $i \geq 1$*,* $\mathtt{He}_i$ *denotes the* i*-th (probabilist's) Hermite polynomial, defined as*

$$\mathtt{He}_i(x) \triangleq (-1)^i \cdot e^{\frac{x^2}{2}} \frac{d^i}{dx^i} e^{-\frac{x^2}{2}}.$$

Then, each i*-th mode of* $\mathfrak{i}_{X;Y}$ *can be written in the standard form as* $\zeta_i(\mathfrak{i}_{X;Y}) = \sigma_i(f_i^* \otimes g_i^*)$*, with*

$$\sigma_i = |\rho|^i, \quad f_i^* = \frac{1}{\sqrt{i!}} \cdot \mathtt{He}_i, \quad g_i^* = \frac{[\operatorname{sgn}(\rho)]^i}{\sqrt{i!}} \cdot \mathtt{He}_i. \tag{116}$$

Finally, $\|\mathfrak{i}_{X;Y}\|^2 = \sum_{i \geq 1} \sigma_i^2 = \dfrac{\rho^2}{1 - \rho^2}$.

The equation (115) is referred to as Mehler's identity or Mehler's decomposition (Mehler, 1866).

In some scenarios, $\mathcal{X}$ and $\mathcal{Y}$ can be of different types, e.g., $\mathcal{X}$ is continuous, and $\mathcal{Y}$ is discrete. A classical example is mixture models, where X and Y corresponds to the observation variable and the latent categorical variable, respectively. Specifically, we introduce the following Gaussian mixture example, which is a multidimensional extension of an example discussed in Buja (1990).

Example 3 *We consider the probability model with* $\mathcal{Y} = \{1, -1\}$*,* $P_Y \equiv \frac{1}{2}$*, and* $\underline{X}|Y = y \sim \mathcal{N}(y \cdot \underline{\mu}, \Sigma)$ *for* $y \in \mathcal{Y}$*, where* $\underline{\mu} \in \mathbb{R}^d$ *and* Σ *is a positive definite matrix of order* d*, for some* $d \geq 1$*. Then, we have* $\operatorname{rank}(\mathfrak{i}_{\underline{X};Y}) = 1$ *with the standard form* $\mathfrak{i}_{\underline{X};Y} = \sigma_1(f_1^* \otimes g_1^*)$*, where* $g_1^*(y) = y$ *and* $f_1^*(\underline{x}) = c \cdot \tanh(\underline{\mu}^{\mathrm{T}} \Sigma^{-1} \underline{x})$ *for some* $c \in \mathbb{R}$*.*

Proof We have

$$\begin{aligned}
\mathfrak{i}_{\underline{X};Y}(\underline{x}, y) &= \frac{p_{\underline{X}|Y}(\underline{x}|y)}{p_{\underline{X}}(\underline{x})} - 1 \\
&= 2 \cdot \frac{\exp(-\frac{1}{2}(\underline{x} - y\underline{\mu})^{\mathrm{T}} \Sigma^{-1} (\underline{x} - y\underline{\mu}))}{\exp(-\frac{1}{2}(\underline{x} + \underline{\mu})^{\mathrm{T}} \Sigma^{-1} (\underline{x} + \underline{\mu})) + \exp(-\frac{1}{2}(\underline{x} - \underline{\mu})^{\mathrm{T}} \Sigma^{-1} (\underline{x} - \underline{\mu}))} - 1 \\
&= \frac{2 \exp(y \underline{\mu}^{\mathrm{T}} \Sigma^{-1} \underline{x})}{\exp(\underline{\mu}^{\mathrm{T}} \Sigma^{-1} \underline{x}) + \exp(-\underline{\mu}^{\mathrm{T}} \Sigma^{-1} \underline{x})} - 1 \\
&= y \cdot \frac{\exp(\underline{\mu}^{\mathrm{T}} \Sigma^{-1} \underline{x}) - \exp(-\underline{\mu}^{\mathrm{T}} \Sigma^{-1} \underline{x})}{\exp(\underline{\mu}^{\mathrm{T}} \Sigma^{-1} \underline{x}) + \exp(-\underline{\mu}^{\mathrm{T}} \Sigma^{-1} \underline{x})} \\
&= y \cdot \tanh(\underline{\mu}^{\mathrm{T}} \Sigma^{-1} \underline{x}),
\end{aligned}$$

which implies that $\operatorname{rank}(\mathfrak{i}_{\underline{X};Y}) = 1$, $f_1^*(\underline{x}) \propto \tanh(\underline{\mu}^{\mathrm{T}} \Sigma^{-1} \underline{x})$, and $g_1^*(y) = y$. ■

From Example 3, the maximal correlation function f_1^* can be represented by a $d \times 1$ linear layer activated by $\tanh(\cdot)$, with zero bias and weight $\Sigma^{-1} \underline{\mu}$.

A.2 Finite Alphabets: Information Vector and Canonical Dependence Matrix

For finite data alphabets, it can be convenient to introduce the vector and matrix representations of features. To begin, we assume the random variable Z takes finite many possible values, i.e., $|\mathcal{Z}| < \infty$, then the resulting feature space $\mathcal{F}_{\mathcal{Z}}$ is a finite-dimensional vector space. It is sometimes more convenient to represent features using vector and matrix notations. Specifically, each $f \in \mathcal{F}_{\mathcal{Z}}$ can be equivalently expressed as one vector in $\mathbb{R}^{|\mathcal{Z}|}$ as follows. Suppose R_Z is the metric distribution, then we can construct an orthonormal basis $\left\{\mathsf{b}_Z^{(z)} \colon z \in \mathcal{Z}\right\}$ of $\mathcal{F}_{\mathcal{Z}}$, where

$$\mathsf{b}_Z^{(z)}(z') \triangleq \frac{\delta_{zz'}}{\sqrt{R_Z(z)}} \quad \text{for all } z, z' \in \mathcal{Z}. \tag{117}$$

We refer to this basis as the *canonical basis* of $\mathcal{F}_{\mathcal{Z}}$. For all $f \in \mathcal{F}_{\mathcal{Z}}$, we can represent f as a linear combination of these basis functions, i.e.,

$$f = \sum_{z' \in \mathcal{Z}} \xi(z') \cdot \mathsf{b}_Z^{(z')},$$

where the coefficient $\xi(z)$ for each $z \in \mathcal{Z}$ is given by

$$\xi(z) = \left\langle f, \mathsf{b}_Z^{(z)} \right\rangle = f(z)\sqrt{R_Z(z)}. \tag{118}$$

In particular, when f is the density ratio $\tilde{\ell}_{P_Z}$ for some $P_Z \in \mathcal{F}_{\mathcal{Z}}$, the corresponding coefficient $\xi(z)$ for each $z \in \mathcal{Z}$ is

$$\xi(z) = \left\langle \tilde{\ell}_{P_Z}, \mathsf{b}_Z^{(z)} \right\rangle = \frac{P_Z(z) - R_Z(z)}{\sqrt{R_Z(z)}}. \tag{119}$$

This establishes a one-to-one correspondence between $\tilde{\ell}_{P_Z}$ (or P_Z) and the vector $\underline{\xi} \triangleq [\xi(z), z \in \mathcal{Z}]^{\mathrm{T}} \in \mathbb{R}^{|\mathcal{Z}|}$, which is referred to as the information vector associated with $\tilde{\ell}_{P_Z}$ (or P_Z).

Similarly, for X and Y with $|\mathcal{X}| < \infty, |\mathcal{Y}| < \infty$, we can represent the CDK function $\mathfrak{i}_{X;Y} \in \mathcal{F}_{\mathcal{X}\times\mathcal{Y}}$ as an $|\mathcal{X}| \times |\mathcal{Y}|$ matrix $\tilde{B}_{X;Y}$:

$$\tilde{B}_{X;Y}(x;y) \triangleq \frac{P_{X,Y}(x,y) - P_X(x)P_Y(y)}{\sqrt{P_X(x)}\sqrt{P_Y(y)}}, \tag{120}$$

which is referred to as the canonical dependence matrix (CDM) of X and Y. With the metric distribution $P_X P_Y$ on $\mathcal{F}_{\mathcal{X}\times\mathcal{Y}}$, each $\tilde{B}_{X;Y}(x;y)$ is the coefficient associated with the basis function $\mathsf{b}_{X,Y}^{(x,y)}$ [cf. (117)].

In addition, we have $\operatorname{rank}(\tilde{B}_{X;Y}) = \operatorname{rank}(\mathfrak{i}_{X;Y})$. Suppose $\sigma_i(f_i^* \otimes g_i^*) = \zeta_i(\mathfrak{i}_{X;Y})$, for each $1 \le i \le \operatorname{rank}(\mathfrak{i}_{X;Y})$, then σ_i is the i-th singular vector of $\tilde{B}_{X;Y}$, and the corresponding i-th left and right singular vector pair are given by $\underline{\xi}_i^X \in \mathbb{R}^{|\mathcal{X}|}, \underline{\xi}_i^Y \in \mathbb{R}^{|\mathcal{Y}|}$, with [cf. (118)]

$$\xi_i^X(x) = \sqrt{P_X(x)} \cdot f_i^*(x), \quad \xi_i^Y(y) = \sqrt{P_Y(y)} \cdot g_i^*(y). \tag{121}$$

Therefore, for small-scale discrete data, we can use the connection (121) to obtain the modal decomposition by solving the SVD of the corresponding CDM.

Appendix B. Characterization of Common Optimal Solutions

We consider optimization problems defined on a common domain $\mathcal{D}$. Specifically, given k functions $h_i\colon \mathcal{D} \to \mathbb{R}$, $i = 1, \ldots, k$, let us consider optimization problems

$$\underset{u \in \mathcal{D}}{\text{maximize}}\, h_i(u), \quad i = 1, \ldots, k. \tag{122}$$

For each $i = 1, \ldots, k$, we denote the optimal solution and optimal value for i-th problem by

$$\mathcal{D}_i^* \triangleq \arg\max_{u \in \mathcal{D}} h_i(u), \quad t_i^* \triangleq \max_{u \in \mathcal{D}} h_i(u), \tag{123}$$

respectively, and suppose $\mathcal{D}_i^* \neq \varnothing, i = 1, \ldots, k$.

Then, the set $\mathcal{D}^* \triangleq \cap_{i=1}^k \mathcal{D}_i^*$ represents the collection of common optimal solutions for all k optimization problems (122). When such common solutions exist, i.e., $\mathcal{D}^*$ is nonempty, we can obtain $\mathcal{D}^*$ by a single optimization program, by using an objective that aggregates original k objectives. We formalize the result as follows.

Proposition 38 *If $\mathcal{D}^* \neq \varnothing$, we have $\mathcal{D}^* = \arg\max_{u \in \mathcal{D}} \Gamma(h_1(u), h_2(u), \ldots, h_k(u))$ for every $\Gamma\colon \mathbb{R}^k \to \mathbb{R}$ that is strictly increasing in each argument.*

Proof Let $\mathcal{D}^{**} \triangleq \arg\max_{u \in \mathcal{D}} h(u)$ with $h(u) \triangleq \Gamma(h_1(u), h_2(u), \ldots, h_k(u))$. Then the proposition is equivalent to $\mathcal{D}^* = \mathcal{D}^{**}$. We then establish $\mathcal{D}^* \subset \mathcal{D}^{**}$ and $\mathcal{D}^{**} \subset \mathcal{D}^*$, respectively.

To prove $\mathcal{D}^* \subset \mathcal{D}^{**}$, take any $u^* \in \mathcal{D}^*$. Then, for all $u \in \mathcal{D}$, we have $h_i(u) \leq h_i(u^*)$, $i = 1, \ldots, k$, which implies that

$$h(u) = \Gamma(h_1(u), \ldots, h_k(u)) \leq \Gamma(h_1(u^*), \ldots, h_k(u^*)) = h(u^*).$$

Therefore, we have $u^* \in \mathcal{D}^{**}$. Since u^* can be arbitrarily chosen from $\mathcal{D}^*$, we have $\mathcal{D}^* \subset \mathcal{D}^{**}$.

We then establish $\mathcal{D}^{**} \subset \mathcal{D}^*$, which is equivalent to

$$(\mathcal{D} \setminus \mathcal{D}^*) \subset (\mathcal{D} \setminus \mathcal{D}^{**}). \tag{124}$$

Note that (124) is trivially true if $\mathcal{D} \setminus \mathcal{D}^* = \varnothing$. Otherwise, take any $u' \in \mathcal{D} \setminus \mathcal{D}^*$. Then, we have $h_i(u') \leq h_i(u^*)$ for all $i \in [k]$, and the strict inequality holds for at least one $i \in [k]$. This implies that

$$h(u') = \Gamma(h_1(u'), \ldots, h_k(u')) < \Gamma(h_1(u^*), \ldots, h_k(u^*)) = h(u^*).$$

Hence, $u' \notin \mathcal{D}^{**}$, and thus $u' \in (\mathcal{D} \setminus \mathcal{D}^{**})$, which establishes (124). ■

To apply Proposition 38, the first step is to test the existence of common optimal solutions. A naive test is to solve all optimization problems (122) and then check the definition, which can be difficult in practice. Instead, we can consider a related multilevel optimization problem as follows. Let $\mathcal{D}_0 \triangleq \mathcal{D}$, and for each $i = 1, \ldots, k$ as, we define $\mathcal{D}_i$ and $t_i \in \mathbb{R}$ as

$$\mathcal{D}_i \triangleq \arg\max_{u \in \mathcal{D}_{i-1}} h_i(u), \quad t_i \triangleq \max_{u \in \mathcal{D}_{i-1}} h_i(u). \tag{125}$$

Note that for each i, the $\mathcal{D}_i$ solved by level i optimization problem gives the elements in $\mathcal{D}_i$ that maximize h_i, and t_i denotes the corresponding optimal value. Therefore, $\mathcal{D}_k$ can be obtained by sequentially solving k optimization problems as defined in (125).

Then, the following result provides an approach to test the existence of common optimal solutions.

Proposition 39 *The following three statements are equivalent:*

1. *The optimization problems* (122) *have common optimal solutions, i.e.,* $\mathcal{D}^* \neq \varnothing$*;*
2. $t_i = t_i^*$ *for all* $i = 1, \ldots, k$*;*
3. $\mathcal{D}_{i-1} \cap \mathcal{D}_i^* \neq \varnothing$ *for all* $i = 1, \ldots, k$*.*

In addition, if one of these statements holds, then we have $\mathcal{D}_k = \mathcal{D}^*$*.*

Proof We establish the equivalence of the statements 1 to 3, by showing that "1" $\Longrightarrow$ "2", "2" $\Longrightarrow$ "3", and "3" $\Longrightarrow$ "1".

"1" $\Longrightarrow$ "2" Suppose "1" holds, and take any $u^* \in \mathcal{D}^* = \cap_{i=1}^k \mathcal{D}_i^*$. We then establish "2" by induction. First, note that $u^* \in \mathcal{D}_0$. For the induction step, we can show that for each $i = 1, \ldots, k$, if $u^* \in \mathcal{D}_{i-1}$, then $u^* \in \mathcal{D}_i$ and $t_i^* = t_i$. Indeed, we have

$$t_i^* \geq t_i = \max_{u \in \mathcal{D}_{i-1}} h_i(u) \geq h_i(u^*) = t_i^*,$$

where the first inequality follows from the fact that $\mathcal{D} = \mathcal{D}_0 \supset \mathcal{D}_1 \supset \cdots \supset \mathcal{D}_k$, where the second inequality follows from the inductive assumption $u^* \in \mathcal{D}_{i-1}$, and where the last equality follows from that $u^* \in \mathcal{D}^* \subset \mathcal{D}_i^*$.

"2" $\Longrightarrow$ "3" For each $i = 2, \ldots, k$, $t_i = t_i^*$ implies that there exists some $u_i \in \mathcal{D}_{i-1}$, such that $h_i(u_i) = t_i = t_i^* = \max_{u \in \mathcal{D}} h_i(u)$, and thus $u_i \in \mathcal{D}_i^*$. Therefore, $u_i \in \mathcal{D}_{i-1} \cap \mathcal{D}_i^*$, which establishes "3".

"3" $\Longrightarrow$ "1" For each $i = 2, \ldots, k$, from $\mathcal{D}_{i-1} \cap \mathcal{D}_i^* \neq \varnothing$ and the definitions (123) and (125), we have $\mathcal{D}_i = \mathcal{D}_{i-1} \cap \mathcal{D}_i^*$. It can also be verified that $\mathcal{D}_i = \mathcal{D}_{i-1} \cap \mathcal{D}_i^*$ holds for $i = 1$. Therefore, we obtain

$$\mathcal{D}_k = \mathcal{D}_{k-1} \cap \mathcal{D}_k^* = \mathcal{D}_{k-2} \cap \mathcal{D}_{k-1}^* \cap \mathcal{D}_k^* = \cdots = \mathcal{D}_0 \cap \left(\bigcap_{i=1}^k \mathcal{D}_i^* \right) = \mathcal{D} \cap \mathcal{D}^* = \mathcal{D}^*. \tag{126}$$

This implies that $\mathcal{D}^* = \mathcal{D}_{k-1} \cap \mathcal{D}_k^* \neq \varnothing$.

Finally, from (126) we know that statement 3 implies $\mathcal{D}_k = \mathcal{D}^*$. Since all three statements are equivalent, each statement implies $\mathcal{D}_k = \mathcal{D}^*$. ■

Appendix C. Proofs

C.1 Proof of Proposition 5

Let $\hat{\gamma} \triangleq \Pi(\gamma; \mathcal{G}_\mathcal{X} \otimes \mathcal{G}_\mathcal{Y})$. We first consider the second equality of (18), i.e.,

$$\zeta_{\leq k}(\hat{\gamma}) = \underset{\substack{\gamma' :\, \gamma' = f \otimes g, \\ f \in \mathcal{G}_\mathcal{X}^k,\, g \in \mathcal{G}_\mathcal{Y}^k}}{\arg\min} \|\gamma - \gamma'\| \tag{127}$$

For each $k \leq \operatorname{rank}(\hat{\gamma})$, consider $\gamma' = f \otimes g$ with $f \in \mathcal{G}_\mathcal{X}^k, g \in \mathcal{G}_\mathcal{Y}^k$. Then, we have

$$\|\gamma - \gamma'\|^2 = \|\gamma - \hat{\gamma} + \hat{\gamma} - \gamma'\|^2 = \|\gamma - \hat{\gamma}\|^2 + \|\hat{\gamma} - \gamma'\|^2 \geq \|\gamma - \hat{\gamma}\|^2 + \|r_k(\hat{\gamma})\|^2, \tag{128}$$

where to obtain the second equality we have used the orthogonality principle with fact that $(\gamma-\hat{\gamma}) \perp \mathcal{G}_{\mathcal{X}} \otimes \mathcal{G}_{\mathcal{Y}}$ and $(\hat{\gamma}-\gamma') \in \mathcal{G}_{\mathcal{X}} \otimes \mathcal{G}_{\mathcal{Y}}$. Note that the inequality in (128) holds with equality if and only if $\gamma' = \zeta_{\leq k}(\hat{\gamma})$, which establishes (127).

We then establish $\zeta_k(\gamma|\mathcal{G}_{\mathcal{X}}, \mathcal{G}_{\mathcal{Y}}) = \zeta_k(\hat{\gamma})$ by induction. To begin, set $k=1$ in (127), and the right hand side becomes $\zeta(\gamma|\mathcal{G}_{\mathcal{X}}, \mathcal{G}_{\mathcal{Y}})$, which implies $\zeta(\gamma|\mathcal{G}_{\mathcal{X}}, \mathcal{G}_{\mathcal{Y}}) = \zeta(\hat{\gamma})$, i.e., $\zeta_1(\gamma|\mathcal{G}_{\mathcal{X}}, \mathcal{G}_{\mathcal{Y}}) = \zeta_1(\hat{\gamma})$. As the inductive hypothesis, suppose we have

$$\zeta_i(\gamma|\mathcal{G}_{\mathcal{X}}, \mathcal{G}_{\mathcal{Y}}) = \zeta_i(\hat{\gamma}), \quad i = 1, \ldots, m. \tag{129}$$

From (17), we have

$$\begin{aligned}
\zeta_{m+1}(\gamma|\mathcal{G}_{\mathcal{X}}, \mathcal{G}_{\mathcal{Y}}) &= \zeta\left(\gamma - \sum_{i=1}^{m} \zeta_i(\gamma|\mathcal{G}_{\mathcal{X}}, \mathcal{G}_{\mathcal{Y}}) \middle| \mathcal{G}_{\mathcal{X}}, \mathcal{G}_{\mathcal{Y}}\right) \\
&= \zeta\left(\gamma - \sum_{i=1}^{m} \zeta_i(\hat{\gamma}) \middle| \mathcal{G}_{\mathcal{X}}, \mathcal{G}_{\mathcal{Y}}\right) && (130) \\
&= \zeta\left(\Pi\left(\gamma - \sum_{i=1}^{m} \zeta_i(\hat{\gamma}); \mathcal{G}_{\mathcal{X}} \otimes \mathcal{G}_{\mathcal{Y}}\right)\right) && (131) \\
&= \zeta\left(\hat{\gamma} - \sum_{i=1}^{m} \Pi\left(\zeta_i(\hat{\gamma}); \mathcal{G}_{\mathcal{X}} \otimes \mathcal{G}_{\mathcal{Y}}\right)\right) && (132) \\
&= \zeta\left(\hat{\gamma} - \sum_{i=1}^{m} \zeta_i(\hat{\gamma})\right) = \zeta_{m+1}(\hat{\gamma}), && (133)
\end{aligned}$$

where (130)–(131) follow from the inductive assumption (129), where (132) follows from the linearity of projection operator. To obtain the first equality of (132), we have again applied the assumption (129): for $i = 1, \ldots, m$, $\zeta_i(\hat{\gamma}) = \zeta_i(\gamma|\mathcal{G}_{\mathcal{X}}, \mathcal{G}_{\mathcal{Y}}) \in \mathcal{G}_{\mathcal{X}} \otimes \mathcal{G}_{\mathcal{Y}}$.

Finally, $\zeta_{\leq k}(\gamma|\mathcal{G}_{\mathcal{X}}, \mathcal{G}_{\mathcal{Y}}) = \zeta_{\leq k}(\hat{\gamma})$ can be readily obtained by definition. ■

C.2 Proof of Proposition 7

It is easy to verify that $\operatorname{cov}(\hat{f}_i^*, \hat{g}_i^*) = \left\langle \mathfrak{i}_{X;Y}, \hat{f}_i^* \otimes \hat{g}_i^* \right\rangle = \| \zeta_i(\mathfrak{i}'_{X;Y}) \| = \hat{\sigma}_i^*$, where $\mathfrak{i}'_{X;Y} = \Pi\left(\mathfrak{i}_{X;Y}; \mathcal{G}_{\mathcal{X}} \otimes \mathcal{G}_{\mathcal{Y}}\right)$.

From Fact 4, we have $(\hat{f}_i^*, \hat{g}_i^*) = \arg\max_{(f,g)\in\hat{\mathcal{D}}'_i} \left\langle \mathfrak{i}'_{X;Y}, f \otimes g \right\rangle$ for each $i \geq 1$, where we have defined each $\hat{\mathcal{D}}'_i$ as

$$\hat{\mathcal{D}}'_i \triangleq \{(f,g) \in \mathcal{F}_{\mathcal{X}} \times \mathcal{F}_{\mathcal{Y}} \colon \|f\| = \|g\| = 1 \text{ and } \langle f, \hat{f}_j^* \rangle = \langle g, \hat{g}_j^* \rangle = 0 \text{ for all } j \in [i-1]\}.$$

Therefore, for each i and $(f,g) \in \hat{\mathcal{D}}_i \subset \hat{\mathcal{D}}'_i$, we have

$$\operatorname{cov}(f_i, g_i) = \langle \mathfrak{i}_{X;Y}, f_i \otimes g_i \rangle = \left\langle \mathfrak{i}'_{X;Y}, f_i \otimes g_i \right\rangle \leq \hat{\sigma}_i^* = \operatorname{cov}(\hat{f}_i^*, \hat{g}_i^*),$$

where the second equality follows from that $f_i \otimes g_i \in \mathcal{G}_{\mathcal{X}} \otimes \mathcal{G}_{\mathcal{Y}}$. Hence, we obtain $(\hat{f}_i^*, \hat{g}_i^*) = \arg\max_{(f,g)\in\hat{\mathcal{D}}_i} \operatorname{cov}(f,g)$. ■

C.3 Proof of Proposition 12

The result of $\|\mathfrak{i}_{X;Y}\|$ directly follows from Property 1. In addition,

$$f^{\mathrm{T}}(x) g(Y) = \mathfrak{i}_{X;Y}(x,y) = \frac{P_{X,Y}(x,y) - P_X(x)P_Y(y)}{P_X(x)P_Y(y)}, \quad \text{for all } (x,y) \in \mathcal{X} \times \mathcal{Y},$$

which implies $P_{Y|X}(y|x) = P_Y(y)\left(1 + f^{\mathrm{T}}(x)g(y)\right)$, i.e., (25).

Therefore, for all $\psi \in \mathcal{F}_{\mathcal{Y}}^d$, we have

$$\begin{aligned}\mathbb{E}\left[\psi(Y)|X=x\right] = \sum_{y\in\mathcal{Y}} P_{Y|X}(y|x)\psi(y) &= \sum_{y\in\mathcal{Y}} P_Y(y)(1+f^{\mathrm{T}}(x)g(y))\psi(y)\\ &= \sum_{y\in\mathcal{Y}} P_Y(y)\psi(y) + \sum_{y\in\mathcal{Y}} P_Y(y)(\psi(y)g^{\mathrm{T}}(y))f(x)\\ &= \mathbb{E}\left[\psi(Y)\right] + \Lambda_{\psi,g}f(x),\end{aligned}$$

which gives (26). ■

C.4 Proof of Proposition 13

We have

$$\begin{aligned}\mathbb{E}\left[\psi(Y)|X=x\right] &= \sum_{y\in\mathcal{Y}} P_{Y|X}(y|x)\psi(y)\\ &= \sum_{y\in\mathcal{Y}} P_Y(y)(1+\mathsf{i}_{X;Y}(x,y))\psi(y)\\ &= \sum_{y\in\mathcal{Y}} P_Y(y)[1+\zeta_{\le k}(\mathsf{i}_{X;Y})](x,y)\psi(y) + \sum_{y\in\mathcal{Y}} P_Y(y)\sum_{i>k}\sigma_i^* f_i^*(x)g_i^*(y)\psi(y)\\ &= \sum_{y\in\mathcal{Y}} P_Y(y)(1+f^{\mathrm{T}}(x)g(y))\psi(y)\\ &= \sum_{y\in\mathcal{Y}} P_Y(y)\psi(y) + \sum_{y\in\mathcal{Y}} P_Y(y)(\psi(y)g^{\mathrm{T}}(y))f(x)\\ &= \mathbb{E}\left[\psi(Y)\right] + \Lambda_{\psi,g}f(x),\end{aligned}$$

where the fourth equality follows from the fact that

$$\sum_{y\in\mathcal{Y}} P_Y(y)\sum_{i>k}\sigma_i^* f_i^*(x)g_i^*(y)\psi(y) = \sum_{i>k}\sigma_i^* f_i^*(x)\mathbb{E}\left[g_i^*(Y)\psi(Y)\right] = 0,$$

due to $\psi_j \in \{\mathtt{1}_{\mathcal{Y}}, g_1^*, \ldots, g_k^*\}$ for each $j \in [d]$. ■

C.5 Proof of Property 3

The property is equivalent to $\mathcal{L}(f,g) = \mathcal{L}(f+\underline{u}, g+\underline{v})$ for all $\underline{u},\underline{v} \in \mathbb{R}^k$. Hence, it suffices to prove that

$$\mathcal{L}(f+\underline{u}, g+\underline{v}) \le \mathcal{L}(f,g), \qquad \text{for all } \underline{u},\underline{v} \in \mathbb{R}^k. \tag{134}$$

Note that for all $b \in \mathcal{F}_{\mathcal{Y}}$, since

$$\left(f(x)+\underline{u}\right)^{\mathrm{T}}\left(g(y)+\underline{v}\right) + b(y) = f(x)\cdot g(y) + \underline{v}^{\mathrm{T}}f(x) + \underline{u}^{\mathrm{T}}g(y) + b(y) + \underline{u}^{\mathrm{T}}\underline{v}, \tag{135}$$

we have [cf. (31)] $\tilde{P}_{Y|X}^{(f+\underline{u},g+\underline{v},b)} = \tilde{P}_{Y|X}^{(f,g,b+\underline{u}^{\mathrm{T}}g)}$, which implies that $\mathcal{L}(f+\underline{u},g+\underline{v},b) = \mathcal{L}(f,g,b+\underline{u}^{\mathrm{T}}g)$.

Therefore, we obtain

$$\mathcal{L}(f+\underline{u},g+\underline{v}) = \max_{b\in\mathcal{F}_{\mathcal{Y}}}\mathcal{L}(f+\underline{u},g+\underline{v},b) \le \max_{b\in\mathcal{F}_{\mathcal{Y}}}\mathcal{L}(f,g,b+\underline{u}^{\mathrm{T}}g) \le \mathcal{L}(f,g). \tag{136}$$

■

C.6 Proof of Proposition 14

We first prove a useful lemma.

Lemma 40 *Suppose $p > 0, q > 0$, $p + q = 1$, then we have*

$$\log\left(p \cdot \exp\left(\frac{u}{p}\right) + q \cdot \exp\left(-\frac{u}{q}\right)\right) \geq \min\left\{u^2, u_0|u|\right\}, \quad \textit{for all } u \in \mathbb{R},$$

where $u_0 \triangleq \frac{\ln 2}{3} \cdot \min\{p, q\}$.

Proof [Proof of Lemma 40] Let $p_{\min} \triangleq \min\{p, q\}$. $h(u) \triangleq \log\left(p \cdot \exp\left(p^{-1}u\right) + q \cdot \exp\left(-q^{-1}u\right)\right)$. Then, we have

$$h'(u) = \frac{\exp\left(p^{-1}u\right) - \exp\left(-q^{-1}u\right)}{p \cdot \exp\left(p^{-1}u\right) + q \cdot \exp\left(-q^{-1}u\right)}$$

$$\begin{aligned}
h''(u) &= \left[p \cdot \exp\left(p^{-1}u\right) + q \cdot \exp\left(-q^{-1}u\right)\right]^{-2} \\
&\quad \cdot \left[\left(\frac{1}{p}\exp\left(\frac{u}{p}\right) + \frac{1}{q}\exp\left(-\frac{u}{q}\right)\right) \cdot \left(p\exp\left(\frac{u}{p}\right) + q\exp\left(-\frac{u}{q}\right)\right) - \left[\exp\left(\frac{u}{p}\right) - \exp\left(-\frac{u}{q}\right)\right]^2\right] \\
&\geq \frac{\left[\exp\left(p^{-1}u\right) + \exp\left(-q^{-1}u\right)\right]^2 - \left[\exp\left(p^{-1}u\right) - \exp\left(-q^{-1}u\right)\right]^2}{\left[p \cdot \exp\left(p^{-1}u\right) + q \cdot \exp\left(-q^{-1}u\right)\right]^2} \\
&= 4 \cdot \frac{\exp\left(\left(p^{-1} - q^{-1}\right) \cdot u\right)}{\left[p \cdot \exp\left(p^{-1}u\right) + q \cdot \exp\left(-q^{-1}u\right)\right]^2}.
\end{aligned}$$

Moreover, for all $|u| \leq u_0$, we have

$$\exp\left(\left(p^{-1} - q^{-1}\right) \cdot u\right) \geq \exp\left(-|p^{-1} - q^{-1}| \cdot u_0\right) \geq \exp\left(-\frac{u_0}{p_{\min}}\right)$$
$$p \cdot \exp\left(p^{-1}u\right) + q \cdot \exp\left(-q^{-1}u\right) \leq \exp\left(\frac{u_0}{p_{\min}}\right).$$

As a result, for all $|u| \leq u_0$, we have $h''(u) \geq 4\exp\left(-\frac{3u_0}{p_{\min}}\right) = 2$.

Therefore, $h'(u_0) \geq h'(0) + 2 \cdot (u_0 - 0) = 2u_0 > u_0$, and similarly, $-h'(-u_0) = h'(0) - h'(-u_0) \geq 2u_0$, i.e., $h'(-u_0) \leq -2u_0 \leq -u_0$. Moreover, for all $|u| \leq u_0$, $h(u) \geq h(0) + h'(0) \cdot u + \frac{1}{2}u^2 \cdot 2 = u^2$. Therefore, for all $u > u_0$, $h'(u) \geq h'(u_0) > u_0$, which implies that $h(u) \geq h(u_0) + u_0(u - u_0) \geq u_0^2 + u_0 u - u_0^2 = u_0 u$. Similarly, we have $h(u) \geq -u_0 u$ for all $u < -u_0$. ■

Proceeding to the proof of Proposition 14, we consider zero-mean k-dimensional f, g. Without loss of generality, we assume $b \in \mathcal{F}_{\mathcal{Y}}$ satisfies $\mathbb{E}\left[b(Y)\right] = -H(Y)$. Then, let $a \in \tilde{\mathcal{F}}_{\mathcal{Y}}$ be $a(y) \triangleq b(y) - \log P_Y(y)$, and define $\gamma \in \mathcal{F}_{\mathcal{X}\times\mathcal{Y}}$ as $\gamma(x, y) \triangleq f(x) \cdot g(y) + a(y)$. Note that since

$$\exp(f(x) \cdot g(y) + b(y)) = P_Y(y)\exp(f(x) \cdot g(y) + a(y)) = P_Y(y)\exp(\gamma(x, y)),$$

we have

$$\tilde{P}^{(f,g,b)}_{Y|X}(y|x) = \frac{\exp(f(x) \cdot g(y) + b(y))}{\sum_{y' \in \mathcal{Y}} \exp(f(x) \cdot g(y') + b(y'))} = \frac{P_Y(y)\exp(\gamma(x, y))}{\sum_{y' \in \mathcal{Y}} P_Y(y')\exp(\gamma(x, y'))}.$$

Therefore,

$$\begin{aligned}
&\mathcal{L}(f,g,b)\\
&\quad= \mathbb{E}_{(\hat{X},\hat{Y})\sim P_X P_Y}\left[(\mathsf{i}_{X;Y}(\hat{X},\hat{Y})+1)\cdot\log\tilde{P}_{Y|X}^{(f,g,b)}(\hat{Y}|\hat{X})\right]\\
&\quad= \mathbb{E}_{(\hat{X},\hat{Y})\sim P_X P_Y}\left[(\mathsf{i}_{X;Y}(\hat{X},\hat{Y})+1)\cdot\left(\log P_Y(\hat{Y})+\gamma(\hat{X},\hat{Y})-\log\sum_{y'\in\mathcal{Y}}P_Y(y')\exp(\gamma(\hat{X},y'))\right)\right]\\
&\quad= -H(Y)+\langle\mathsf{i}_{X;Y},\gamma\rangle-\mathbb{E}_{\hat{X}\sim P_X}\left[\log\sum_{y'\in\mathcal{Y}}P_Y(y')\exp(\gamma(\hat{X},y'))\right]. \qquad (137)
\end{aligned}$$

As a result, for all (f,g,b) that satisfies $\mathcal{L}(f,g,b)\geq -H(Y)$, we have

$$\langle\mathsf{i}_{X;Y},\gamma\rangle\geq\mathbb{E}_{\hat{X}\sim P_X}\left[\log\sum_{y'\in\mathcal{Y}}P_Y(y')\exp(\gamma(\hat{X},y'))\right]. \qquad (138)$$

In addition, note that for all $x\in\mathcal{X}$ and $y\in\mathcal{Y}$, we have

$$\begin{aligned}
\sum_{y'\in\mathcal{Y}}P_Y(y')\exp(\gamma(x,y')) &= P_Y(y)\exp(\gamma(x,y))+(1-P_Y(y))\sum_{y'\in\mathcal{Y}\setminus\{y\}}\frac{P_Y(y')}{1-P_Y(y)}\exp(\gamma(x,y'))\\
&\geq P_Y(y)\exp(\gamma(x,y))+(1-P_Y(y))\exp\left(-\frac{P_Y(y)}{1-P_Y(y)}\cdot\gamma(x,y)\right).
\end{aligned}$$

where the inequality follows from Jensen's inequality and $\sum_{y\in\mathcal{Y}}P_Y(y)\gamma(x,y)=0$.

Let us define

$$q_X\triangleq\min_{x\in\mathcal{X}}P_X(x)>0,\quad q_Y\triangleq\min_{y\in\mathcal{Y}}P_Y(y)>0.$$

Then, from Lemma 40 we have

$$\begin{aligned}
\log\sum_{y'\in\mathcal{Y}}P_Y(y')\exp(\gamma(x,y')) &\geq \log\left[P_Y(y)\exp(\gamma(x,y))+(1-P_Y(y))\exp\left(-\frac{P_Y(y)}{1-P_Y(y)}\cdot\gamma(x,y)\right)\right]\\
&\geq \min\left\{(P_Y(y)\gamma(x,y))^2,\frac{\ln 2}{3}\cdot q_Y|P_Y(y)\gamma(x,y)|\right\}\\
&\geq \frac{\ln 2\cdot q_Y^2}{3}\cdot\min\left\{(\gamma(x,y))^2,|\gamma(x,y)|\right\},
\end{aligned}$$

which implies that

$$\begin{aligned}
\mathbb{E}_{\hat{X}\sim P_X}\left[\log\sum_{y'\in\mathcal{Y}}P_Y(y')\exp(\gamma(\hat{X},y'))\right] &\geq P_X(x)\cdot\log\sum_{y'\in\mathcal{Y}}P_Y(y')\exp(\gamma(x,y'))\\
&\geq \frac{\ln 2\cdot q_X q_Y^2}{3}\cdot\min\left\{(\gamma(x,y))^2,|\gamma(x,y)|\right\}. \qquad (139)
\end{aligned}$$

On the other hand, since $\|\mathsf{i}_{X;Y}\|=O(\epsilon)$, there exists a constant $C>0$ such that $\|\mathsf{i}_{X;Y}\|\leq C\epsilon$. Therefore,

$$\langle\mathsf{i}_{X;Y},\gamma\rangle\leq\|\mathsf{i}_{X;Y}\|\cdot\|\gamma\|\leq\gamma_{\max}\cdot\epsilon, \qquad (140)$$

where $\gamma_{\max} \triangleq \max_{x\in\mathcal{X},y\in\mathcal{Y}} |\gamma(x,y)|$.

Hence, by combining (138), (139) and (140), we obtain

$$\frac{\ln 2 \cdot q_X q_Y^2}{3} \cdot \min\left\{\gamma_{\max}^2, \gamma_{\max}\right\} \le C \cdot \gamma_{\max} \cdot \epsilon, \tag{141}$$

where we have taken $(x,y) = \arg\max_{x',y'} |\gamma(x',y')|$ in (139).

From (141), if $\epsilon < \frac{\ln 2}{3} \cdot \frac{q_X q_Y^2}{3C}$ we have $\gamma_{\max} < \epsilon$. This implies that

$$|\gamma(x,y)| < \epsilon, \quad \text{for all } x \in \mathcal{X}, y \in \mathcal{Y}, \tag{142}$$

and we obtain $\|\gamma\| < \epsilon$. In addition, note that since $\|\gamma\|^2 = \|f \otimes g + a\|^2 = \|f \otimes g\|^2 + \|a\|^2$, we obtain $\|f \otimes g\| = O(\epsilon)$.

From (142), we have

$$\begin{aligned}
\sum_{y'\in\mathcal{Y}} P_Y(y') \exp(\gamma(x,y')) &= \sum_{y'\in\mathcal{Y}} P_Y(y') \left(1 + \gamma(x,y') + \frac{(\gamma(x,y'))^2}{2} + o(\epsilon^2)\right) \\
&= 1 + \frac{1}{2} \sum_{y'\in\mathcal{Y}} P_Y(y') (\gamma(x,y'))^2 + o(\epsilon^2) \\
&= 1 + \frac{1}{2} \cdot \mathbb{E}_{\hat{Y}\sim P_Y}\left[\gamma(x,\hat{Y})\right]^2 + o(\epsilon^2).
\end{aligned}$$

Therefore,

$$\mathbb{E}_{\hat{X}\sim P_X}\left[\log \sum_{y'\in\mathcal{Y}} P_Y(y') \exp(\gamma(\hat{X},y'))\right] = \frac{1}{2} \cdot \mathbb{E}_{(\hat{X},\hat{Y})\sim P_X P_Y}\left[\gamma(\hat{X},\hat{Y})\right]^2 + o(\epsilon^2) = \frac{1}{2} \cdot \|\gamma\|^2 + o(\epsilon^2),$$

and the likelihood (137) becomes

$$\begin{aligned}
\mathcal{L}(f,g,b) &= -H(Y) + \langle \mathfrak{i}_{X;Y}, \gamma\rangle - \mathbb{E}_{\hat{X}\sim P_X}\left[\log \sum_{y'\in\mathcal{Y}} P_Y(y') \exp(\gamma(\hat{X},y'))\right] \\
&= -H(Y) + \langle \mathfrak{i}_{X;Y}, \gamma\rangle - \frac{1}{2} \cdot \|\gamma\|^2 + o(\epsilon^2) \\
&= \frac{1}{2} \cdot \left(\|\mathfrak{i}_{X;Y}\|^2 - \|\mathfrak{i}_{X;Y} - \gamma\|^2\right) - H(Y) + o(\epsilon^2) \\
&= \frac{1}{2} \cdot \left(\|\mathfrak{i}_{X;Y}\|^2 - \|\mathfrak{i}_{X;Y} - f\otimes g - a\|^2\right) - H(Y) + o(\epsilon^2) \\
&= \frac{1}{2} \cdot \left(\|\mathfrak{i}_{X;Y}\|^2 - \|\mathfrak{i}_{X;Y} - f\otimes g\|^2 - \|a\|^2\right) - H(Y) + o(\epsilon^2),
\end{aligned} \tag{143}$$

where the last equality follows from the fact that

$$\|\mathfrak{i}_{X;Y} - f\otimes g - a\|^2 = \|\mathfrak{i}_{X;Y} - f\otimes g\|^2 + \|a\|^2,$$

due to the orthogonality $(\mathfrak{i}_{X;Y} - f\otimes g) \perp \mathcal{F}_{\mathcal{Y}} \ni a$.

Finally, from (143), for given f, g, $\mathcal{L}(f,g,b)$ is maximized when $\|a\| = o(\epsilon)$. Therefore, we have

$$\mathcal{L}(f,g) = \max_{b\in\mathcal{F}_{\mathcal{Y}}} \mathcal{L}(f,g,b) = \frac{1}{2} \cdot \left(\|\mathfrak{i}_{X;Y}\|^2 - \|\mathfrak{i}_{X;Y} - f\otimes g\|^2\right) - H(Y) + o(\epsilon^2),$$

which gives (33). ■

C.7 Proof of Theorem 19

Throughout this proof, we consider

$$L_1(\bar{f},\bar{g},f,g) \triangleq \left\|\mathfrak{i}_{X;Y} - \bar{f}\otimes\bar{g}\right\|^2, \quad L_2(\bar{f},\bar{g},f,g) \triangleq \left\|\mathfrak{i}_{X;Y} - \bar{f}\otimes\bar{g} - f\otimes g\right\|^2$$

defined on the domain $\mathcal{D} \triangleq \{(\bar{f},\bar{g},f,g)\colon \bar{f}\in\mathcal{G}_{\mathcal{X}}^{\bar{k}}, \bar{g}\in\mathcal{F}_{\mathcal{Y}}^{\bar{k}}, f\in\mathcal{F}_{\mathcal{X}}^{k}, g\in\mathcal{F}_{\mathcal{Y}}^{k}\}$, and let

$$\mathcal{D}_1^* \triangleq \operatorname*{arg\,min}_{(\bar{f},\bar{g},f,g)\in\mathcal{D}} L_1(\bar{f},\bar{g},f,g), \quad \mathcal{D}_2^* \triangleq \operatorname*{arg\,min}_{(\bar{f},\bar{g},f,g)\in\mathcal{D}} L_2(\bar{f},\bar{g},f,g).$$

We also define $\bar{\gamma},\gamma\in\mathcal{F}_{\mathcal{X}\times\mathcal{Y}}$, as

$$\bar{\gamma} \triangleq \Pi\left(\mathfrak{i}_{X;Y};\mathcal{G}_{\mathcal{X}}\otimes\mathcal{F}_{\mathcal{Y}}\right), \quad \gamma \triangleq \mathfrak{i}_{X;Y} - \bar{\gamma} = \Pi\left(\mathfrak{i}_{X;Y};(\mathcal{F}_{\mathcal{X}}\boxminus\mathcal{G}_{\mathcal{X}})\otimes\mathcal{F}_{\mathcal{Y}}\right). \tag{144}$$

Then, it suffices to establish that

$$\mathcal{D}^* \triangleq \mathcal{D}_1^*\cap\mathcal{D}_2^* = \{(\bar{f},\bar{g},f,g)\in\mathcal{D}\colon \bar{f}\otimes\bar{g} = \bar{\gamma}, f\otimes g = \zeta_{\leq k}(\gamma)\}. \tag{145}$$

To see this, note that the common solution $\mathcal{D}^*$ (145) coincides with the optimal solution (47) to be established. From Proposition 38, since $\mathcal{D}^*\neq\varnothing$, we have

$$\mathcal{D}^* = \operatorname*{arg\,min}_{(\bar{f},\bar{g},f,g)\in\mathcal{D}} L_1(\bar{f},\bar{g},f,g) + L_2(\bar{f},\bar{g},f,g). \tag{146}$$

Furthermore, from the definition of the H-score (cf. Definition 10), we have

$$\begin{aligned}
\mathscr{H}\left(\begin{bmatrix}\bar{f}\\ f\end{bmatrix},\begin{bmatrix}\bar{g}\\ g\end{bmatrix};\mathcal{C}_\pi\right) &= \mathscr{H}(\bar{f},\bar{g}) + \mathscr{H}\left(\begin{bmatrix}\bar{f}\\ f\end{bmatrix},\begin{bmatrix}\bar{g}\\ g\end{bmatrix}\right)\\
&= \frac{1}{2}\cdot\left(\|\mathfrak{i}_{X;Y}\|^2 - L_1(\bar{f},\bar{g},f,g)\right) + \frac{1}{2}\cdot\left(\|\mathfrak{i}_{X;Y}\|^2 - L_2(\bar{f},\bar{g},f,g)\right)\\
&= \|\mathfrak{i}_{X;Y}\|^2 - \frac{1}{2}\left(L_1(\bar{f},\bar{g},f,g) + L_2(\bar{f},\bar{g},f,g)\right),
\end{aligned}$$

which implies that

$$\operatorname*{arg\,max}_{(\bar{f},\bar{g},f,g)\in\mathcal{D}} \mathscr{H}\left(\begin{bmatrix}\bar{f}\\ f\end{bmatrix},\begin{bmatrix}\bar{g}\\ g\end{bmatrix};\mathcal{C}_\pi\right) = \operatorname*{arg\,min}_{(\bar{f},\bar{g},f,g)\in\mathcal{D}} L_1(\bar{f},\bar{g},f,g) + L_2(\bar{f},\bar{g},f,g) = \mathcal{D}^*.$$

It remains only to establish (145). From Proposition 39, it suffices to establish that $t_2 = t_2^*$ and (cf. (145))

$$\mathcal{D}_2 = \{(\bar{f},\bar{g},f,g)\in\mathcal{D}\colon \bar{f}\otimes\bar{g} = \bar{\gamma}, f\otimes g = \gamma\}, \tag{147}$$

where we have defined

$$t_2^* \triangleq \min_{(\bar{f},\bar{g},f,g)\in\mathcal{D}} L_2(\bar{f},\bar{g},f,g), \quad t_2 \triangleq \min_{(\bar{f},\bar{g},f,g)\in\mathcal{D}_1^*} L_2(\bar{f},\bar{g},f,g), \quad \mathcal{D}_2 \triangleq \operatorname*{arg\,min}_{(\bar{f},\bar{g},f,g)\in\mathcal{D}_1^*} L_2(\bar{f},\bar{g},f,g).$$

To this end, let us define $\bar{h}\in\mathcal{G}_{\mathcal{X}}^k$ and $\bar{h}\in(\mathcal{F}_{\mathcal{X}}\boxminus\mathcal{G}_{\mathcal{X}})^k$ as $\bar{h}\triangleq\Pi\left(f;\mathcal{G}_{\mathcal{X}}\right)$, and $h\triangleq f-\bar{h} = \Pi\left(f;\mathcal{F}_{\mathcal{X}}\boxminus\mathcal{G}_{\mathcal{X}}\right)$. Then, we have

$$L_2(\bar{f},\bar{g},f,g) = \left\|\mathfrak{i}_{X;Y} - \bar{f}\otimes\bar{g} - f\otimes g\right\|^2 \tag{148}$$

$$= \left\|\bar{\gamma} + \gamma - \bar{f} \otimes \bar{g} - (\bar{h} + h) \otimes g\right\|^2 \tag{149}$$

$$= \left\|\bar{\gamma} - \bar{f} \otimes \bar{g} - \bar{h} \otimes g\right\|^2 + \left\|\gamma - h \otimes g\right\|^2 \tag{150}$$

$$\geq \left\|\gamma - h \otimes g\right\|^2 \tag{151}$$

$$\geq \left\|r_k(\gamma)\right\|^2 \tag{152}$$

where (150) follows from the orthogonality principle, since $\left(\bar{\gamma} - \bar{f} \otimes \bar{g} - \bar{h} \otimes g\right) \in \mathcal{G}_{\mathcal{X}} \otimes \mathcal{F}_{\mathcal{Y}}$ and $(\gamma - h \otimes g) \perp \mathcal{G}_{\mathcal{X}} \otimes \mathcal{F}_{\mathcal{Y}}$. Moreover, it is easily verified that the lower bound (152) is tight: all inequalities hold with equality for $(\bar{f}, \bar{g}, f, g)$ satisfying (47).

Therefore, we have

$$t_2^* = \min_{(\bar{f},\bar{g},f,g)\in\mathcal{D}} L_2(\bar{f}, \bar{g}, f, g) = \left\|r_k(\gamma)\right\|^2 \tag{153}$$

On the other hand, since $\bar{k} \geq \operatorname{rank}(\bar{\gamma})$, from Proposition 5, we have $\mathcal{D}_1^* = \{(\bar{f}, \bar{g}, f, g) \colon \bar{f} \otimes \bar{g} = \bar{\gamma}\}$. Hence, for all $(\bar{f}, \bar{g}, f, g) \in \mathcal{D}_1^*$, we have

$$L_2(\bar{f}, \bar{g}, f, g) = \left\|\mathfrak{i}_{X;Y} - \bar{f} \otimes \bar{g} - f \otimes g\right\|^2 = \left\|\mathfrak{i}_{X;Y} - \bar{\gamma} - f \otimes g\right\|^2 = \left\|\gamma - f \otimes g\right\|^2 \geq \left\|r_k(\gamma)\right\|^2,$$

where the inequality holds with equality if and only if $f \otimes g = \zeta_{\leq k}(\gamma)$.

As a result, $\mathcal{D}_2 = \arg\min_{(\bar{f},\bar{g},f,g)\in\mathcal{D}_1^*} L_2(\bar{f}, \bar{g}, f, g)$ is given by (147), and we have

$$t_2 = \min_{(\bar{f},\bar{g},f,g)\in\mathcal{D}_1^*} L_2(\bar{f}, \bar{g}, f, g) = \left\|r_k(\gamma)\right\|^2 = t_2^*.$$

■

C.8 Proof of Theorem 20

To begin, we have

$$\begin{aligned}
\mathscr{H}\left(\begin{bmatrix}\bar{f}\\ f\end{bmatrix}, \begin{bmatrix}\bar{g}\\ g\end{bmatrix}; \mathfrak{C}_\pi^\star\right) &= \sum_{i=1}^{\bar{k}} \mathscr{H}(\bar{f}_{[i]}, \bar{g}_{[i]}) + \sum_{i=1}^{k} \mathscr{H}\left(\begin{bmatrix}\bar{f}\\ f_{[i]}\end{bmatrix}, \begin{bmatrix}\bar{g}\\ g_{[i]}\end{bmatrix}\right) \\
&= \sum_{i=1}^{\bar{k}-1} \mathscr{H}(\bar{f}_{[i]}, \bar{g}_{[i]}) + \sum_{i=1}^{k} \left(k^{-1} \cdot \mathscr{H}(\bar{f}, \bar{g}) + \mathscr{H}\left(\begin{bmatrix}\bar{f}\\ f_{[i]}\end{bmatrix}, \begin{bmatrix}\bar{g}\\ g_{[i]}\end{bmatrix}\right)\right).
\end{aligned}$$

Note that for $\left(\begin{bmatrix}\bar{f}\\ f\end{bmatrix}, \begin{bmatrix}\bar{g}\\ g\end{bmatrix}\right) \in \operatorname{dom}(\mathfrak{C}_\pi^\star)$ and each $i \in [\bar{k}]$, $\mathscr{H}(\bar{f}_{[i]}, \bar{g}_{[i]})$ is maximized if and only if

$$\bar{f}_{[i]} \otimes \bar{g}_{[i]} = \zeta_{\leq i}(\mathfrak{i}_{X;Y} | \mathcal{G}_{\mathcal{X}}, \mathcal{F}_{\mathcal{Y}}). \tag{154}$$

In addition, from Theorem 19, for each $i \in [k]$, the term $\left(k^{-1} \cdot \mathscr{H}(\bar{f}, \bar{g}) + \mathscr{H}\left(\begin{bmatrix}\bar{f}\\ f_{[i]}\end{bmatrix}, \begin{bmatrix}\bar{g}\\ g_{[i]}\end{bmatrix}\right)\right)$ is maximized if and only if

$$\bar{f} \otimes \bar{g} = \Pi\left(\mathfrak{i}_{X;Y}; \mathcal{G}_{\mathcal{X}} \otimes \mathcal{F}_{\mathcal{Y}}\right), \qquad f_{[i]} \otimes g_{[i]} = \zeta_{\leq i}(\Pi\left(\mathfrak{i}_{X;Y}; (\mathcal{F}_{\mathcal{X}} \boxminus \mathcal{G}_{\mathcal{X}}) \otimes \mathcal{F}_{\mathcal{Y}}\right)). \tag{155}$$

Note that (48) is the common solution of (154) and (155). Hence, the proof is completed by applying Proposition 38. ■

C.9 Proof of Proposition 22

The relation $\mathfrak{i}_{X;S} = \tilde{\ell}_{P^{\mathsf{M}}_{X,S,Y}}$ can be directly verified from definition. To establish $\pi_{\mathsf{M}}(\mathfrak{i}_{X;S,Y}) = \Pi\left(\mathfrak{i}_{X;S,Y}; \mathcal{F}_{\mathcal{X}\times\mathcal{S}}\right) = \mathfrak{i}_{X;S}$, from Fact 2, it suffices to show that $(\mathfrak{i}_{X;S,Y} - \mathfrak{i}_{X;S}) \perp \mathcal{F}_{\mathcal{X}\times\mathcal{S}}$.

To this end, note that

$$\begin{aligned}\mathfrak{i}_{X;S,Y}(x,s,y) - \mathfrak{i}_{X;S}(x,s) &= \frac{P_{X,S,Y}(x,s,y) - P^{\mathsf{M}}_{X,S,Y}(x,s,y)}{R_{X,S,Y}(x,s,y)} \\ &= \frac{P_{X,S,Y}(x,s,y) - P_{X|S}(x|s)P_S(s)P_{Y|S}(y|s)}{R_{X,S,Y}(x,s,y)}.\end{aligned}$$

Therefore, for all $f \in \mathcal{F}_{\mathcal{X}\times\mathcal{S}}$, we have

$$\begin{aligned}\left\langle \mathfrak{i}_{X;S,Y} - \mathfrak{i}_{X;S}, f\right\rangle &= \sum_{x\in\mathcal{X},s\in\mathcal{S},y\in\mathcal{Y}} P_{X,S,Y}(x,s,y)f(x,s) - \sum_{x\in\mathcal{X},s\in\mathcal{S},y\in\mathcal{Y}} P_{X|S}(x|s)P_S(s)P_{Y|S}(y|s)\cdot f(x,s) \\ &= \mathbb{E}_{P_{X,S}}\left[f(X,S)\right] - \mathbb{E}_{P_{X,S}}\left[f(X,S)\right] \\ &= 0.\end{aligned}$$

■

C.10 Proof of Proposition 23

Let $\gamma = \mathfrak{i}^{(Q)}_{X;S,Y}$, then the statement is equivalent to

$$\gamma \in \mathcal{F}_{\mathcal{X}\times\mathcal{S}} \quad \Longleftrightarrow \quad Q_{X,S,Y} = Q_{X|S}Q_SQ_{Y|S}.$$

"$\Longrightarrow$" If $\gamma \in \mathcal{F}_{\mathcal{X}\times\mathcal{S}}$, then we have

$$\begin{aligned}Q_{X,S,Y}(x,s,y) &= Q_X(x)Q_{S,Y}(s,y)\left(1+\gamma(x,s)\right) \\ &= P_X(x)P_{S,Y}(s,y)\left(1+\gamma(x,s)\right) \\ &= P_X(x)P_S(s)\left(1+\gamma(x,s)\right)\cdot P_{Y|S}(y|s), \quad \text{for all } x,s,y. \end{aligned} \tag{156}$$

Therefore,

$$Q_{X,S}(x,s) = \sum_{y\in\mathcal{Y}} Q_{X,S,Y}(x,s,y) = P_X(x)P_S(s)\left(1+\gamma(x,s)\right). \tag{157}$$

From (156) and (157), we obtain

$$Q_{X,S,Y}(x,s,y) = Q_{X,S}(x,s)P_{Y|S}(y|s) = Q_{X,S}(x,s)Q_{Y|S}(y|s) = Q_{X|S}(x|s)Q_S(s)Q_{Y|S}(y|s). \tag{158}$$

"$\Longleftarrow$" It suffices to note that

$$\mathfrak{i}^{(Q)}_{X;S,Y}(x,s,y) = \frac{Q_{X,S,Y}(x,s,y)}{Q_X(x)Q_{S,Y}(s,y)} - 1 = \frac{Q_{X,S}(x,s)Q_{Y|S}(y|s)}{Q_X(x)Q_S(s)Q_{Y|S}(y|s)} - 1 = \frac{Q_{X,S}(x,s)}{Q_X(x)Q_S(s)} - 1.$$

■

C.11 Proof of Proposition 25

The results on $\|\mathfrak{i}_{X;S}\|$ and $\|\mathfrak{i}_{X;Y|S}\|$ directly follows from Property 1. To establish (63), note that from Proposition 22, we have

$$P_{X,S,Y}(x,s,y) - P_{X|S}(x|s)P_S(s)P_{Y|S}(y|s) = P_X(x)P_{S,Y}(s,y) \cdot \mathfrak{i}_{X;Y|S}(x,s,y),$$

which implies that

$$\begin{aligned} P_{Y|X,S}(y|x,s) &= P_{Y|S}(y|s) \cdot \left(1 + \frac{P_X(x)P_S(s)}{P_{X,S}(x,s)} \cdot \mathfrak{i}_{X;Y|S}(x,s,y)\right) \\ &= P_{Y|S}(y|s) \cdot \left(1 + \frac{1}{1+\mathfrak{i}_{X;S}(x,s)} \cdot \mathfrak{i}_{X;Y|S}(x,s,y)\right) \qquad (159) \\ &= P_{Y|S}(y|s) \cdot \left(1 + \frac{f^{\mathrm{T}}(x)g(s,y)}{1+\bar{f}^{\mathrm{T}}(x)\bar{g}(s)}\right), \qquad (160) \end{aligned}$$

which further implies (64). ■

C.12 Proof of Proposition 26

When X and (S,Y) are ϵ-dependent, we have $\|\mathfrak{i}_{X;S,Y}\| = O(\epsilon)$, and thus

$$\mathfrak{i}_{X;S}(x,s) = O(\epsilon), \quad \mathfrak{i}_{X;Y|S}(x,s,y) = O(\epsilon).$$

Therefore, we have

$$\frac{1}{1+\mathfrak{i}_{X;S}(x,s)} \cdot \mathfrak{i}_{X;Y|S}(x,s,y) = (1-\mathfrak{i}_{X;S}(x,s)) \cdot \mathfrak{i}_{X;Y|S}(x,s,y) + o(\epsilon) = \mathfrak{i}_{X;Y|S}(x,s,y) + o(\epsilon).$$

Then, it follows from (159) that

$$P_{Y|X,S}(y|x,s) = P_{Y|S}(y|s) \cdot \big(1 + \mathfrak{i}_{X;Y|S}(x,s,y)\big) + o(\epsilon).$$

Finally, the proof is completed by using the decomposition (66). ■

C.13 Proof of Theorem 27

For each s,x,y, let us define $R^{(s)}_{X,Y}(x,y) \triangleq P_{X|S=s}(x)P_{Y|S=s}(y)$ and

$$\mathfrak{i}^{(s)}_{X;Y}(x,y) \triangleq \frac{P_{X,Y|S=s}(x,y) - P_{X|S=s}(x)P_{Y|S=s}(y)}{P_{X|S=s}(x)P_{Y|S=s}(y)}. \qquad (161)$$

In addition, we define $\underline{\mu}_s \in \mathbb{R}^k$ as $\underline{\mu}_s \triangleq \mathbb{E}\,[f(X)|S=s]$ for each $s \in \mathcal{S}$. Also, let $\beta \in \mathcal{F}_{\mathcal{S}\times\mathcal{Y}}$ be defined as $\beta(s,y) \triangleq \underline{\mu}_s^{\mathrm{T}} g(s,y)$.

Then, for each $s \in \mathcal{S}$, from (72) and Proposition 14 we have

$$|f^{\mathrm{T}}(x)g(s,y) - \beta(s,y)| = |(f(x)-\underline{\mu}_s)^{\mathrm{T}} g(s,y)| = O(\epsilon), \quad \text{for all } x,s,y, \qquad (162)$$

which implies that

$$\big|f^{\mathrm{T}}(x)g(s,y)\big| = O(\epsilon), \quad \text{for all } x,s,y. \qquad (163)$$

In addition, from Property 3, we have

$$\begin{aligned}
&\mathcal{L}_S^{(s)}(f, g^{(s)}) \\
&\quad = \mathcal{L}_S^{(s)}(f - \underline{\mu}_s, g^{(s)}) \\
&\quad = \frac{1}{2} \cdot \left(\left\| \mathfrak{i}_{X;Y}^{(s)} \right\|_{R_{X,Y}^{(s)}}^2 - \left\| \mathfrak{i}_{X;Y}^{(s)} - (f - \underline{\mu}_s) \otimes g^{(s)} \right\|_{R_{X,Y}^{(s)}}^2 \right) - H(Y|S=s) + o(\epsilon^2) \\
&\quad = \left\langle \mathfrak{i}_{X;Y}^{(s)}, f \otimes g^{(s)} - \underline{\mu}_s^{\mathrm{T}} g^{(s)} \right\rangle_{R_{X,Y}^{(s)}} - \frac{1}{2} \cdot \left\| f \otimes g^{(s)} - \underline{\mu}_s^{\mathrm{T}} g^{(s)} \right\|_{R_{X,Y}^{(s)}}^2 - H(Y|S=s) + o(\epsilon^2), \quad (164)
\end{aligned}$$

where $\langle \cdot, \cdot \rangle_R$ and $\| \cdot \|_R$ denote the inner product and corresponding induced norm on the function space, with respect to the metric distribution R.

For the first two terms in (164), we compute their expectations over P_S. For the first term,

$$\begin{aligned}
&\sum_{s \in \mathcal{S}} P_S(s) \left\langle \mathfrak{i}_{X;Y}^{(s)}, f \otimes g^{(s)} - \underline{\mu}_s^{\mathrm{T}} g^{(s)} \right\rangle_{R_{X,Y}^{(s)}} \\
&\quad = \sum_{x \in \mathcal{X}, s \in \mathcal{S}, y \in \mathcal{Y}} P_S(s) R_{X,Y}^{(s)}(x,y) \mathfrak{i}_{X;Y}^{(s)}(x,y) \left(f^{\mathrm{T}}(x) g^{(s)}(y) - \underline{\mu}_s^{\mathrm{T}} g^{(s)}(y) \right) \\
&\quad = \sum_{x \in \mathcal{X}, s \in \mathcal{S}, y \in \mathcal{Y}} P_X(x) P_{S,Y}(s,y) \mathfrak{i}_{X;Y|S}(x,s,y) \cdot \left(f^{\mathrm{T}}(x) g(s,y) - \beta(s,y) \right) \\
&\quad = \left\langle \mathfrak{i}_{X;Y|S}, f \otimes g \right\rangle - \left\langle \mathfrak{i}_{X;Y|S}, \beta \right\rangle \\
&\quad = \left\langle \mathfrak{i}_{X;Y|S}, f \otimes g \right\rangle \quad (165)
\end{aligned}$$

where to obtain the second equality we have used the facts that

$$\begin{aligned}
\mathfrak{i}_{X;Y}^{(s)}(x,y) &= \frac{P_{X,Y|S=s}(x,y) - P_{X|S=s}(x) P_{Y|S=s}(y)}{P_{X|S=s}(x) P_{Y|S=s}(y)} \\
&= \frac{P_{X,S,Y}(x,s,y) - P_{X,S,Y}^{\mathsf{M}}(x,s,y)}{P_X(x) P_{S,Y}(s,y)} \cdot \frac{P_X(x)}{P_{X|S=s}(x)} \\
&= \mathfrak{i}_{X;Y|S}(x,s,y) \cdot \frac{1}{1 + \mathfrak{i}_{X;S}(x,s)} \quad (166)
\end{aligned}$$

and

$$P_S(s) R_{X,Y}^{(s)}(x,y) = P_{X,S,Y}^{\mathsf{M}}(x,s,y) = P_X(x) P_{S,Y}(s,y) \cdot (1 + \mathfrak{i}_{X;S}(x,s)), \quad (167)$$

and where to obtain (165) we have used the fact that $\mathfrak{i}_{X;Y|S} \perp \mathcal{F}_{\mathcal{S} \times \mathcal{Y}} \ni \beta$.

For the second term of (164), we have

$$\begin{aligned}
&\sum_{s \in \mathcal{S}} P_S(s) \left\| f \otimes g^{(s)} - \underline{\mu}_s^{\mathrm{T}} g^{(s)} \right\|_{R_{X,Y}^{(s)}}^2 \\
&\quad = \sum_{x \in \mathcal{X}, s \in \mathcal{S}, y \in \mathcal{Y}} P_{X,S,Y}^{\mathsf{M}}(x,s,y) \cdot \left[(f(x) - \underline{\mu}_s)^{\mathrm{T}} g(s,y) \right]^2 \quad (168) \\
&\quad = \sum_{x \in \mathcal{X}, s \in \mathcal{S}, y \in \mathcal{Y}} P_X(x) P_{S,Y}(s,y) \cdot \left[(f(x) - \underline{\mu}_s)^{\mathrm{T}} g(s,y) \right]^2 + o(\epsilon^2) \quad (169) \\
&\quad = \sum_{x \in \mathcal{X}, s \in \mathcal{S}, y \in \mathcal{Y}} P_X(x) P_{S,Y}(s,y) \cdot \left[f^{\mathrm{T}}(x) g(s,y) - \beta(s,y) \right]^2 + o(\epsilon^2) \quad (170)
\end{aligned}$$

$$= \|f \otimes g - \beta\|^2 + o(\epsilon^2) \tag{171}$$
$$= \|f \otimes g\|^2 + \|\beta\|^2 + o(\epsilon^2), \tag{172}$$

where to obtain the second equality we have used (162), (167), and the fact that $\|\mathfrak{i}_{X;S}\| = O(\epsilon)$. Furthermore, we can show that $\|\beta\| = o(\epsilon)$. To see this, note that

$$\begin{aligned}\beta(s,y) = \underline{\mu}_s^{\mathrm{T}} g(s,y) &= \sum_{x\in\mathcal{X}} P_{X|S}(x|s) f^{\mathrm{T}}(x) g(s,y) \\ &= \sum_{x\in\mathcal{X}} P_X(x)\cdot[1+\mathfrak{i}_{X;S}(x,s)]\cdot f^{\mathrm{T}}(x) g(s,y) \\ &= \sum_{x\in\mathcal{X}} P_X(x)\cdot \mathfrak{i}_{X;S}(x,s)\cdot f^{\mathrm{T}}(x) g(s,y).\end{aligned}$$

Then, from $\|\mathfrak{i}_{X;S}\| = O(\epsilon)$ and (163), we obtain $|\beta(s,y)| = O(\epsilon^2)$ and $\|\beta\| = O(\epsilon^2) = o(\epsilon)$.

Therefore, we can refine (172) as

$$\sum_{s\in\mathcal{S}} P_S(s) \big\| f\otimes g^{(s)} - \underline{\mu}_s^{\mathrm{T}} g^{(s)} \big\|^2_{R^{(s)}_{X,Y}} = \|f\otimes g\|^2 + o(\epsilon^2). \tag{173}$$

Combining (164), (165), and (173), we have

$$\begin{aligned}\mathcal{L}_S(f,g) = \sum_{s\in\mathcal{S}} P_S(s)\cdot \mathcal{L}_S^{(s)}(f,g^{(s)}) &= \big\langle \mathfrak{i}_{X;Y|S}, f\otimes g\big\rangle - \frac{1}{2}\cdot \big\|f\otimes g\big\|^2 - H(Y|S) + o(\epsilon^2) \\ &= \frac{1}{2}\cdot\Big(\|\mathfrak{i}_{X;Y|S}\|^2 - \big\|\mathfrak{i}_{X;Y|S} - f\otimes g\big\|^2\Big) - H(Y|S) + o(\epsilon^2),\end{aligned}$$

which is maximized if and only if $f\otimes g = \zeta_{\le k}(\mathfrak{i}_{X;Y|S}) + o(\epsilon)$. ■

C.14 Proof of Proposition 28

Given any $h \in \mathcal{F}_{\mathcal{X}_1\times\mathcal{Y}} + \mathcal{F}_{\mathcal{X}_2\times\mathcal{Y}}$, we can represent $h = h^{(1)} + h^{(2)}$ with $h^{(i)} \in \mathcal{F}_{\mathcal{X}_i\times\mathcal{Y}}, i = 1,2$, i.e., $h(x_1,x_2,y) = h^{(1)}(x_1,y) + h^{(2)}(x_2,y)$.

Then, for any $Q_{X_1,X_2,Y} \in \mathcal{Q}_{\mathsf{B}}$, we have

$$\begin{aligned}\langle \mathfrak{i}^{(Q)}_{X_1,X_2;Y}, h^{(i)}\rangle &= \sum_{x_1\in\mathcal{X}_1, x_2\in\mathcal{X}_2, y\in\mathcal{Y}} P_{X_1,X_2}(x_1,x_2) P_Y(y) \mathfrak{i}^{(Q)}_{X_1,X_2;Y}(x_1,x_2,y)\cdot h^{(i)}(x_i,y) \\ &= \sum_{x_1\in\mathcal{X}_1, x_2\in\mathcal{X}_2, y\in\mathcal{Y}} [Q_{X_1,X_2,Y}(x_1,x_2,y) - P_{X_1,X_2}(x_1,x_2)P_Y(y)]\cdot h^{(i)}(x_i,y) \\ &= \sum_{x_i\in\mathcal{X}_i, y\in\mathcal{Y}} [P_{X_i,Y}(x_i,y) - P_{X_i}(x_i)P_Y(y)]\cdot h^{(i)}(x_i,y) \\ &= \langle \mathfrak{i}_{X_i;Y}, h^{(i)}\rangle \qquad\qquad \text{for } i=1,2.\end{aligned}$$

As a result,

$$\begin{aligned}\langle \pi_{\mathsf{B}}(\mathfrak{i}^{(Q)}_{X_1,X_2;Y}), h\rangle &= \langle \mathfrak{i}^{(Q)}_{X_1,X_2;Y} - \pi_{\mathsf{I}}(\mathfrak{i}^{(Q)}_{X_1,X_2;Y}), h\rangle \\ &= \langle \mathfrak{i}^{(Q)}_{X_1,X_2;Y}, h\rangle \\ &= \langle \mathfrak{i}^{(Q)}_{X_1,X_2;Y}, h^{(1)}\rangle + \langle \mathfrak{i}^{(Q)}_{X_1,X_2;Y}, h^{(2)}\rangle \equiv \langle \mathfrak{i}_{X_1;Y}, h^{(1)}\rangle + \langle \mathfrak{i}_{X_2;Y}, h^{(2)}\rangle,\end{aligned}$$

where the second equality follows from the fact that $\langle \pi_{\mathsf{I}}(\mathfrak{i}^{(Q)}_{X_1,X_2;Y}), h\rangle = 0$.

Hence, we obtain $\langle \pi_{\mathsf{B}}(\mathfrak{i}^{(Q)}_{X_1,X_2;Y}) - \pi_{\mathsf{B}}(\tilde{\ell}_{P_{X_1,X_2,Y}}), h\rangle = 0$ for all $h \in \mathcal{F}_{\mathcal{X}_1\times\mathcal{Y}} + \mathcal{F}_{\mathcal{X}_2\times\mathcal{Y}}$, which implies that $\pi_{\mathsf{B}}(\mathfrak{i}^{(Q)}_{X_1,X_2;Y}) - \pi_{\mathsf{B}}(\tilde{\ell}_{P_{X_1,X_2,Y}}) = 0$. ■

C.15 Proof of Proposition 30

From the definition, we have

$$\begin{aligned}P_{X_1,X_2,Y}(x_1,x_2,y) &= P_{X_1,X_2}(x_1,x_2)P_Y(y)\left(1+[\pi_{\mathsf{B}}(\mathfrak{i}_{X_1,X_2;Y})](x_1,x_2,y)+[\pi_{\mathsf{I}}(\mathfrak{i}_{X_1,X_2;Y})](x_1,x_2,y)\right)\\ &= P_{X_1,X_2}(x_1,x_2)P_Y(y)\left[1+\bar{f}^{\mathrm{T}}(x_1,x_2)\bar{g}(y)+f^{\mathrm{T}}(x_1,x_2)g(y)\right],\end{aligned}$$

which implies (83a).

To obtain (83b), note that from (174), we have

$$\begin{aligned}P_{X_1,Y}(x_1,y) &= \sum_{x_2\in\mathcal{X}_2} P^{\mathsf{B}}_{X_1,X_2,Y}(x_1,x_2,y)\\ &= P_{X_1}(x_1)P_Y(y)\left[1+\mathbb{E}\left[\bar{f}^{\mathrm{T}}(x_1,X_2)\bar{g}(y)|X_1=x_1\right]\right]\\ &= P_{X_1}(x_1)P_Y(y)\left[1+\left(\bar{f}^{(1)}(x_1)+[\tau_1(\bar{f}^{(2)})](x_1)\right)^{\mathrm{T}}\bar{g}(y)\right],\end{aligned}$$

where to obtain the last equality we have used the fact that $\tau_1(\bar{f}) = \bar{f}^{(1)} + \tau_1(\bar{f}^{(2)})$. Similarly, we can obtain (83b).

Finally, (84) can be readily obtained from (83). ■

C.16 Proof of Proposition 31

Note that since $\mathfrak{i}_{X_1,X_2;Y} \in \tilde{\mathcal{F}}_{\mathcal{X}_1\times\mathcal{X}_2\times\mathcal{Y}}$, we have $\pi_{\mathsf{B}}(\mathfrak{i}_{X_1,X_2;Y}) \in \tilde{\mathcal{F}}_{\mathcal{X}_1\times\mathcal{X}_2\times\mathcal{Y}}$, which implies that

$$\sum_{(x_1,x_2,y)\in\mathcal{X}_1\times\mathcal{X}_2\times\mathcal{Y}} P_{X_1,X_2}(x_1,x_2)P_Y(y)\cdot[\pi_{\mathsf{B}}(\mathfrak{i}_{X_1,X_2;Y})](x_1,x_2,y) = \langle \pi_{\mathsf{B}}(\mathfrak{i}_{X_1,X_2;Y}), 1\rangle = 0.$$

Therefore, it follows from the definition of $P^{\mathsf{B}}_{X_1,X_2,Y}$ that $\sum_{(x_1,x_2,y)\in\mathcal{X}_1\times\mathcal{X}_2\times\mathcal{Y}} P^{\mathsf{B}}_{X_1,X_2,Y}(x_1,x_2,y) = 1$. Similarly, we have $\sum_{(x_1,x_2,y)\in\mathcal{X}_1\times\mathcal{X}_2\times\mathcal{Y}} P^{\mathsf{I}}_{X_1,X_2,Y}(x_1,x_2,y) = 1$.

Moreover, from (85), we have $P^{\mathsf{B}}_{X_1,X_2,Y}(x_1,x_2,y) \geq 0, P^{\mathsf{I}}_{X_1,X_2,Y}(x_1,x_2,y) \geq 0$ for all (x_1,x_2,y). Therefore, we obtain $P^{\mathsf{B}}_{X_1,X_2,Y}, P^{\mathsf{I}}_{X_1,X_2,Y} \in \mathcal{P}^{\mathcal{X}_1\times\mathcal{X}_2\times\mathcal{Y}}$.

Since $\mathfrak{i}_{X_1,X_2;Y} \perp \mathcal{F}_{\mathcal{X}_1\times\mathcal{X}_2}$, we have $\pi_{\mathsf{B}}(\mathfrak{i}_{X_1,X_2;Y}) \perp \mathcal{F}_{\mathcal{X}_1\times\mathcal{X}_2}$. Therefore, for any $(\hat{x}_1,\hat{x}_2) \in \mathcal{X}_1\times\mathcal{X}_2$, let $f(x_1,x_2) = \delta_{x_1\hat{x}_1}\delta_{x_2\hat{x}_2}$, then we have

$$\begin{aligned}\sum_{y\in\mathcal{Y}}[\pi_{\mathsf{B}}(\mathfrak{i}_{X_1,X_2;Y})](\hat{x}_1,\hat{x}_2,y) &= \sum_{(x_1,x_2,y)\in\mathcal{X}_1\times\mathcal{X}_2\times\mathcal{Y}}[\pi_{\mathsf{B}}(\mathfrak{i}_{X_1,X_2;Y})](x_1,x_2,y)\cdot\delta_{x_1\hat{x}_1}\delta_{x_2\hat{x}_2}\\ &= \langle\pi_{\mathsf{B}}(\mathfrak{i}_{X_1,X_2;Y}), f\rangle = 0.\end{aligned}$$

This implies that $P^{\mathsf{B}}_{X_1,X_2} = P_{X_1,X_2}$. Similarly, we have $P^{\mathsf{I}}_{X_1,X_2} = P_{X_1,X_2}$.

Finally, note that since

$$P_{X_1,X_2,Y}(x_1,x_2,y) = P_{X_1,X_2}(x_1,x_2)P_Y(y)\left[1+[\pi_{\mathsf{B}}(\mathfrak{i}_{X_1,X_2;Y})](x_1,x_2,y)+[\pi_{\mathsf{I}}(\mathfrak{i}_{X_1,X_2;Y})](x_1,x_2,y)\right],$$

we have

$$\begin{aligned}P_{X_1,Y}(x_1,y) - P^{\mathsf{B}}_{X_1,Y}(x_1,y) &= P^{\mathsf{I}}_{X_1,Y}(x_1,y) - P_{X_1}(x_1)P_Y(y)\\ &= \sum_{x_2'\in\mathcal{X}_2} P_{X_1,X_2}(x_1,x_2')P_Y(y)[\pi_{\mathsf{I}}(\mathfrak{i}_{X_1,X_2;Y})](x_1,x_2',y)\end{aligned}$$

$$= 0.$$

To obtain the last equality, note that for any $\hat{x}_1, \hat{y} \in \mathcal{X}_1 \times \mathcal{Y}$, let us define $\gamma \in \mathcal{F}_{\mathcal{X}_1 \times \mathcal{Y}}$ as $\gamma(x_1, y) = \delta_{x_1 \hat{x}_1} \delta_{y\hat{y}}$. Then, from $\pi_{\mathsf{I}}(\mathfrak{i}_{X_1,X_2;Y}) \perp \mathcal{F}_{\mathcal{X}_1 \times \mathcal{Y}}$, we have

$$0 = \langle \pi_{\mathsf{I}}(\mathfrak{i}_{X_1,X_2;Y}), \gamma \rangle = \sum_{x_2' \in \mathcal{X}_2} P_{X_1,X_2}(\hat{x}_1, x_2') P_Y(\hat{y}) [\pi_{\mathsf{I}}(\mathfrak{i}_{X_1,X_2;Y})](\hat{x}_1, x_2', \hat{y}). \tag{174}$$

Similarly, we can show that

$$P_{X_2,Y}(x_2, y) - P^{\mathsf{B}}_{X_2,Y}(x_2, y) = P^{\mathsf{I}}_{X_2,Y}(x_2, y) - P_{X_2}(x_2) P_Y(y) = 0.$$

■

C.17 Proof of Proposition 32

Note that since

$$H(Q_{X_1,X_2,Y}) = H(P_{X_1,X_2}) + H(P_Y) - I_Q(X_1, X_2; Y),$$

we have

$$P^{\mathsf{ent}}_{X_1,X_2,Y} = \operatorname*{arg\,min}_{Q_{X_1,X_2,Y} \in \mathcal{Q}_{\mathsf{B}}} I_Q(X_1, X_2; Y), \tag{175}$$

where $I_Q(X_1, X_2; Y)$ denotes the mutual information between (X_1, X_2) and Y with respect to $Q_{X_1,X_2,Y}$.

Specifically, when we take $P_{X_1,X_2,Y}$ as the $Q_{X_1,X_2,Y}$, we have $I_P(X_1, X_2; Y) = \frac{1}{2} \cdot \|\mathfrak{i}_{X_1,X_2;Y}\|^2 + o(\epsilon^2)$. Therefore, to solve (175), it suffices to consider $Q_{X_1,X_2,Y}$ with $I_Q(X_1, X_2; Y) < I_P(X_1, X_2; Y) = O(\epsilon^2)$. Since $Q_{X_1,X_2} Q_Y = P_{X_1,X_2} P_Y \in \operatorname{relint}(\mathcal{P}^{\mathcal{X}_1 \times \mathcal{X}_2 \times \mathcal{Y}})$, we have $\|\tilde{\ell}_{Q_{X_1,X_2,Y}}\|^2 = O(\epsilon^2)$ [cf. (Sason and Verdú, 2016, Eq. (338))].

Therefore,

$$\begin{aligned}
I_Q(X_1, X_2; Y) &= \frac{1}{2} \cdot \|\tilde{\ell}_{Q_{X_1,X_2,Y}}\|^2 + o(\epsilon^2) \\
&= \frac{1}{2} \cdot \|\pi_{\mathsf{B}}(\tilde{\ell}_{Q_{X_1,X_2,Y}})\|^2 + \frac{1}{2} \cdot \|\pi_{\mathsf{I}}(\tilde{\ell}_{Q_{X_1,X_2,Y}})\|^2 + o(\epsilon^2) \\
&= \frac{1}{2} \cdot \|\pi_{\mathsf{B}}(\mathfrak{i}_{X_1,X_2;Y})\|^2 + \frac{1}{2} \cdot \|\pi_{\mathsf{I}}(\tilde{\ell}_{Q_{X_1,X_2,Y}})\|^2 + o(\epsilon^2) \\
&\geq \frac{1}{2} \cdot \|\pi_{\mathsf{B}}(\mathfrak{i}_{X_1,X_2;Y})\|^2 + o(\epsilon^2)
\end{aligned}$$

where the inequality holds with equality when $\left\|\pi_{\mathsf{I}}(\tilde{\ell}_{Q_{X_1,X_2,Y}})\right\| = o(\epsilon)$. Hence, we obtain $\mathfrak{i}^{(\mathsf{ent})}_{X_1,X_2;Y} = \tilde{\ell}_{P^{\mathsf{ent}}_{X_1,X_2,Y}} = \pi_{\mathsf{B}}(\mathfrak{i}_{X_1,X_2;Y}) + o(\epsilon)$, which gives (88). ■

C.18 Proof of Proposition 33

For all $\phi \triangleq \phi^{(1)} + \phi^{(2)} \in \mathcal{F}_{\mathcal{X}_1} + \mathcal{F}_{\mathcal{X}_2}$ and $\psi \in \mathcal{F}_{\mathcal{Y}}$ with $\|\psi\| = 1$, we have $\mathbb{E}_{P_{X_1,X_2} P_Y}[\psi(Y)\phi(X_1, X_2)] = 0$. Hence,

$$\mathbb{E}[\psi(Y)\phi(X_1, X_2)] = \mathbb{E}_{P_{X_1,X_2} P_Y}[(1 + \mathfrak{i}_{X_1,X_2;Y}(X_1, X_2, Y)) \cdot \psi(Y)\phi(X_1, X_2)]$$

$$
\begin{aligned}
&= \langle \mathbf{i}_{X_1,X_2;Y}, \phi \otimes \psi \rangle \\
&= \langle \pi_{\mathsf{B}}(\mathbf{i}_{X_1,X_2;Y}), \phi \otimes \psi \rangle + \langle \pi_{\mathsf{I}}(\mathbf{i}_{X_1,X_2;Y}), \phi \otimes \psi \rangle \\
&= \langle \pi_{\mathsf{B}}(\mathbf{i}_{X_1,X_2;Y}), \phi \otimes \psi \rangle,
\end{aligned}
$$

where the first equality follows from the definition of $\mathbf{i}_{X_1,X_2;Y}$, and where the last equality follow from the fact that $\pi_{\mathsf{I}}(\mathbf{i}_{X_1,X_2;Y}) \perp (\mathcal{F}_{\mathcal{X}_1 \times \mathcal{Y}} + \mathcal{F}_{\mathcal{X}_2 \times \mathcal{Y}}) \ni \phi \otimes \psi$.

Therefore, we can rewrite the objective (92) as

$$
\begin{aligned}
\mathbb{E}\left[\Big(\psi(Y) - \phi^{(1)}(X_1) - \phi^{(2)}(X_2)\Big)^2\right] &= \mathbb{E}\left[(\psi(Y) - \phi(X_1, X_2))^2\right] \\
&= \|\psi\|^2 + \|\phi\|^2 - 2 \cdot \mathbb{E}\left[\psi(Y)\phi(X_1, X_2)\right] \\
&= 1 + \|\phi \otimes \psi\|^2 - 2\langle \pi_{\mathsf{B}}(\mathbf{i}_{X_1,X_2;Y}), \phi \otimes \psi \rangle \\
&= 1 + \|\pi_{\mathsf{B}}(\mathbf{i}_{X_1,X_2;Y}) - \phi \otimes \psi\|^2 - \|\pi_{\mathsf{B}}(\mathbf{i}_{X_1,X_2;Y})\|^2.
\end{aligned}
$$

Finally, the proof is completed by noting that $\|\pi_{\mathsf{B}}(\mathbf{i}_{X_1,X_2;Y})\|^2 = \sum_{i=1}^{\bar{K}} \bar{\sigma}_i^2$, and we have

$$
\|\pi_{\mathsf{B}}(\mathbf{i}_{X_1,X_2;Y}) - \phi \otimes \psi\|^2 \geq \|r_1(\pi_{\mathsf{B}}(\mathbf{i}_{X_1,X_2;Y}))\|^2 = \sum_{i=2}^{\bar{K}} \bar{\sigma}_i^2,
$$

where the inequality becomes equality if and only if $\phi \otimes \psi = \zeta(\pi_{\mathsf{B}}(\mathbf{i}_{X_1,X_2;Y})) = \bar{\sigma}_1(\bar{f}_1^* \otimes \bar{g}_1^*)$.

■

C.19 Proof of Theorem 34

We first introduce a useful lemma.

Lemma 41 *For all $k \geq 1, f \in \mathcal{F}_{\mathcal{X}}^k, g \in \mathcal{F}_{\mathcal{Y}}^k$, let $\bar{h} \triangleq \Pi(f; \mathcal{F}_{\mathcal{X}_1} + \mathcal{F}_{\mathcal{X}_2})$, $h = f - \bar{h}$. Then, we have*

$$
\mathscr{H}_{\mathrm{m}}(f, g) = \frac{1}{2} \cdot \left[L(R_{X_1,X_2,Y}) - \eta_0 \cdot \big\| \pi_{\mathsf{I}}(\tilde{\ell}_{\hat{P}^{(0)}_{X_1,X_2,Y}}) - h \otimes g \big\|^2 - L_{\mathsf{B}}(\bar{h} \otimes g) \right], \tag{176}
$$

where we have defined

$$
L_{\mathsf{B}}(\gamma) \triangleq \eta_0 \cdot \big\| \pi_{\mathsf{B}}(\tilde{\ell}_{\hat{P}^{(0)}_{X_1,X_2,Y}}) - \pi_{\mathsf{B}}(\gamma) \big\|^2 + \eta_1 \cdot \big\| \tilde{\ell}_{\hat{P}^{(1)}_{X_1,Y}} - \pi_{\mathsf{M}_1}(\gamma) \big\|^2 + \eta_2 \cdot \big\| \tilde{\ell}_{\hat{P}^{(2)}_{X_2,Y}} - \pi_{\mathsf{M}_2}(\gamma) \big\|^2. \tag{177}
$$

Proof By definition, we have

$$
\begin{aligned}
\mathscr{H}(f, g; \hat{P}^{(0)}_{X_1,X_2,Y}) &= \frac{1}{2} \left(\big\| \tilde{\ell}_{\hat{P}^{(0)}_{X_1,X_2,Y}} \big\|^2 - \big\| \tilde{\ell}_{\hat{P}^{(0)}_{X_1,X_2,Y}} - f \otimes g \big\|^2 \right) \\
&= \frac{1}{2} \left(\big\| \tilde{\ell}_{\hat{P}^{(0)}_{X_1,X_2,Y}} \big\|^2 - \big\| (\pi_{\mathsf{I}}(\tilde{\ell}_{\hat{P}^{(0)}_{X_1,X_2,Y}}) - h \otimes g) + (\pi_{\mathsf{B}}(\tilde{\ell}_{\hat{P}^{(0)}_{X_1,X_2,Y}}) - \bar{h} \otimes g) \big\|^2 \right) \\
&= \frac{1}{2} \left(\big\| \tilde{\ell}_{\hat{P}^{(0)}_{X_1,X_2,Y}} \big\|^2 - \big\| \pi_{\mathsf{I}}(\tilde{\ell}_{\hat{P}^{(0)}_{X_1,X_2,Y}}) - h \otimes g \big\|^2 - \big\| \pi_{\mathsf{B}}(\tilde{\ell}_{\hat{P}^{(0)}_{X_1,X_2,Y}}) - \bar{h} \otimes g \big\|^2 \right),
\end{aligned} \tag{178}
$$

where to obtain the last equality we have used the orthogonality between $\left(\pi_{\mathsf{I}}(\tilde{\ell}_{\hat{P}^{(0)}_{X_1,X_2,Y}}) - h \otimes g \right) \perp (\mathcal{F}_{\mathcal{X}_1 \times \mathcal{Y}} + \mathcal{F}_{\mathcal{X}_2 \times \mathcal{Y}})$ and $\left(\pi_{\mathsf{B}}(\tilde{\ell}_{\hat{P}^{(0)}_{X_1,X_2,Y}}) - \bar{h} \otimes g \right) \in (\mathcal{F}_{\mathcal{X}_1 \times \mathcal{Y}} + \mathcal{F}_{\mathcal{X}_2 \times \mathcal{Y}})$.

In addition, for each $i = 1, 2$, from $\tau_i(f) = \tau_i(\bar{h})$ we have

$$\begin{aligned}\mathscr{H}(\tau_i(f), g; \hat{P}^{(i)}_{X_i,Y}) = \mathscr{H}(\tau_i(\bar{h}), g; \hat{P}^{(i)}_{X_i,Y}) &= \frac{1}{2}\left(\big\|\tilde{\ell}_{\hat{P}^{(i)}_{X_i,Y}}\big\|^2 - \big\|\tilde{\ell}_{\hat{P}^{(i)}_{X_i,Y}} - \tau_i(\bar{h}) \otimes g\big\|^2\right), \\ &= \frac{1}{2}\left(\big\|\tilde{\ell}_{\hat{P}^{(i)}_{X_i,Y}}\big\|^2 - \big\|\tilde{\ell}_{\hat{P}^{(i)}_{X_i,Y}} - \pi_{\mathsf{M}_i}(\bar{h} \otimes g)\big\|^2\right), \end{aligned} \tag{179}$$

where the last equality follows from that $\pi_{\mathsf{M}_i}(\bar{h} \otimes g) = \tau_i(\bar{h}) \otimes g$.

From (178) and (179), we can rewrite (95) as

$$\begin{aligned}\mathscr{H}_{\mathrm{m}}(f, g) &= \eta_0 \cdot \mathscr{H}(f, g; \hat{P}^{(0)}_{X_1,X_2,Y}) + \eta_1 \cdot \mathscr{H}\left(\tau_1(f), g; \hat{P}^{(1)}_{X_1,Y}\right) + \eta_2 \cdot \mathscr{H}\left(\tau_2(f), g; \hat{P}^{(2)}_{X_2,Y}\right) \\ &= \frac{\eta_0}{2} \cdot \left(\big\|\tilde{\ell}_{\hat{P}^{(0)}_{X_1,X_2,Y}}\big\|^2 - \big\|\pi_{\mathsf{I}}(\tilde{\ell}_{\hat{P}^{(0)}_{X_1,X_2,Y}}) - h \otimes g\big\|^2 - \big\|\pi_{\mathsf{B}}(\tilde{\ell}_{\hat{P}^{(0)}_{X_1,X_2,Y}}) - \bar{h} \otimes g\big\|^2\right) \\ &\quad + \sum_{i=1}^{2} \frac{\eta_i}{2} \cdot \left(\big\|\tilde{\ell}_{\hat{P}^{(i)}_{X_i,Y}}\big\|^2 - \big\|\tilde{\ell}_{\hat{P}^{(i)}_{X_1,Y}} - \pi_{\mathsf{M}_i}(\bar{h} \otimes g)\big\|^2\right) \\ &= \frac{1}{2} \cdot \left[L(R_{X_1,X_2,Y}) - \eta_0 \cdot \big\|\pi_{\mathsf{I}}(\tilde{\ell}_{\hat{P}^{(0)}_{X_1,X_2,Y}}) - h \otimes g\big\|^2\right] \\ &\quad - \frac{1}{2}\left[\eta_0 \cdot \big\|\pi_{\mathsf{B}}(\tilde{\ell}_{\hat{P}^{(0)}_{X_1,X_2,Y}}) - \bar{h} \otimes g\big\|^2 + \sum_{i=1}^{2} \eta_i \cdot \big\|\tilde{\ell}_{\hat{P}^{(i)}_{X_1,Y}} - \pi_{\mathsf{M}_i}(\bar{h} \otimes g)\big\|^2\right] \\ &= \frac{1}{2} \cdot \left[L(R_{X_1,X_2,Y}) - \eta_0 \cdot \big\|\pi_{\mathsf{I}}(\tilde{\ell}_{\hat{P}^{(0)}_{X_1,X_2,Y}}) - h \otimes g\big\|^2 - L_{\mathsf{B}}(\bar{h} \otimes g)\right], \end{aligned}$$

where the last equality follows from the fact that $\bar{h} \otimes g = \pi_{\mathsf{B}}(\bar{h} \otimes g)$.

■

Let use define $\bar{h} \triangleq \Pi\left(f; \mathcal{F}_{\mathcal{X}_1} + \mathcal{F}_{\mathcal{X}_2}\right)$ and $h \triangleq f - \bar{h}$. Then, from Lemma 41, we have

$$\mathscr{H}_{\mathrm{m}}(\bar{f}, \bar{g}) = \frac{1}{2} \cdot \left[L(R_{X_1,X_2,Y}) - \eta_0 \cdot \big\|\pi_{\mathsf{I}}(\tilde{\ell}_{\hat{P}^{(0)}_{X_1,X_2,Y}})\big\|^2 - L_{\mathsf{B}}(\bar{f} \otimes \bar{g})\right], \tag{180}$$

and, similarly,

$$\mathscr{H}_{\mathrm{m}}\left(\begin{bmatrix}\bar{f} \\ f\end{bmatrix}, \begin{bmatrix}\bar{g} \\ g\end{bmatrix}\right) = \frac{1}{2} \cdot \left[L(R_{X_1,X_2,Y}) - \eta_0 \cdot \big\|\pi_{\mathsf{I}}(\tilde{\ell}_{\hat{P}^{(0)}_{X_1,X_2,Y}}) - h \otimes g\big\|^2 - L_{\mathsf{B}}(\bar{f} \otimes \bar{g} + \bar{h} \otimes g)\right]. \tag{181}$$

Therefore, (180) is minimized if and only if

$$\bar{f} \otimes \bar{g} = \pi_{\mathsf{B}}\Big(\tilde{\ell}_{P^{(\mathrm{est})}_{X_1,X_2,Y}}\Big), \tag{182}$$

and (181) is minimized if and only if

$$\bar{f} \otimes \bar{g} + \bar{h} \otimes g = \pi_{\mathsf{B}}\Big(\tilde{\ell}_{P^{(\mathrm{est})}_{X_1,X_2,Y}}\Big), \qquad h \otimes g = \zeta_{\leq k}\left(\pi_{\mathsf{I}}\Big(\tilde{\ell}_{\hat{P}^{(0)}_{X_1,X_2,Y}}\Big)\right). \tag{183}$$

Hence, the common solution of (182) and (183) is

$$\bar{f} \otimes \bar{g} = \pi_{\mathsf{B}}\Big(\tilde{\ell}_{P^{(\mathrm{est})}_{X_1,X_2,Y}}\Big), \qquad \bar{h} \otimes g = 0, \qquad h \otimes g = \zeta_{\leq k}\left(\pi_{\mathsf{I}}\Big(\tilde{\ell}_{\hat{P}^{(0)}_{X_1,X_2,Y}}\Big)\right), \tag{184}$$

which is equivalent to (97). Finally, from Proposition 38, this is also the solution that maximizes the nested H-score (96). ■

C.20 Proof of Theorem 35

We first introduce two useful facts.

Fact 6 (Cover and Thomas 2006, Theorem 11.1.2) *Given m samples $Z_1, \ldots, Z_m$ i.i.d. generated from $Q_Z \in \mathcal{P}^{\mathcal{Z}}$, th probability of observing $\{Z_i\}_{i=1}^m = \{z_i\}_{i=1}^m$, denoted by $\mathbb{P}\{\{z_i\}_{i=1}^m; Q_Z\}$, satisfies $-\log \mathbb{P}\{\{z_i\}_{i=1}^m; Q_Z\} = m\left[H(\hat{P}_Z) + D(\hat{P}_Z \| Q_Z)\right]$, where $\hat{P}_Z$ is the empirical distribution of $\{z_i\}_{i=1}^m$.*

Fact 7 (Huang et al. 2024, Lemma 4.5) *Given a reference distribution $R \in \operatorname{relint}(\mathcal{P}^{\mathcal{Z}})$. Then, for $P, Q \in \mathcal{P}^{\mathcal{Z}}$ with $\|\tilde{\ell}_{P;R}\| = O(\epsilon)$, $\|\tilde{\ell}_{Q;R}\| = O(\epsilon)$, we have*

$$D(P\|Q) = \frac{1}{2} \cdot \|\tilde{\ell}_{P;R} - \tilde{\ell}_{Q;R}\|^2 + o(\epsilon^2).$$

Note that since the data from three different datasets are generated independently, we have

$$\begin{aligned}\mathbb{P}\{\mathscr{D}_0, \mathscr{D}_1, \mathscr{D}_2; Q_{X_1,X_2,Y}\} &= \mathbb{P}\{\mathscr{D}_0; Q_{X_1,X_2,Y}\} \cdot \mathbb{P}\{\mathscr{D}_1; Q_{X_1,X_2,Y}\} \cdot \mathbb{P}\{\mathscr{D}_2; Q_{X_1,X_2,Y}\} \\ &= \mathbb{P}\{\mathscr{D}_0; Q_{X_1,X_2,Y}\} \cdot \mathbb{P}\{\mathscr{D}_1; Q_{X_1,Y}\} \cdot \mathbb{P}\{\mathscr{D}_2; Q_{X_2,Y}\}.\end{aligned}$$

Therefore, from Fact 6,

$$\begin{aligned}&-\log \mathbb{P}\{\mathscr{D}_0, \mathscr{D}_1, \mathscr{D}_2; Q_{X_1,X_2,Y}\} \\ &\quad= -\log \mathbb{P}\{\mathscr{D}_0; Q_{X_1,X_2,Y}\} - \log \mathbb{P}\{\mathscr{D}_1; Q_{X_1,Y}\} - \log \mathbb{P}\{\mathscr{D}_2; Q_{X_2,Y}\} \\ &\quad= n_0 \cdot D\big(\hat{P}^{(0)}_{X_1,X_2,Y} \big\| Q_{X_1,X_2,Y}\big) + n_1 \cdot D\big(\hat{P}^{(1)}_{X_1,Y} \big\| Q_{X_1,Y}\big) + n_2 \cdot D\big(\hat{P}^{(2)}_{X_2,Y} \big\| Q_{X_2,Y}\big) \\ &\qquad\quad + n_0 \cdot H(\hat{P}^{(0)}_{X_1,X_2,Y}) + n_1 \cdot H(\hat{P}^{(1)}_{X_1,Y}) + n_2 \cdot H(\hat{P}^{(2)}_{X_2,Y}) \\ &\quad= n \cdot L^{(\mathrm{ML})}(Q_{X_1,X_2,Y}) + n_0 \cdot H(\hat{P}^{(0)}_{X_1,X_2,Y}) + n_1 \cdot H(\hat{P}^{(1)}_{X_1,Y}) + n_2 \cdot H(\hat{P}^{(2)}_{X_2,Y})\end{aligned}$$

where the second equality follows from Fact 6, and where we have defined

$$L^{(\mathrm{ML})}(Q_{X_1,X_2,Y}) \triangleq \eta_0 \cdot D\big(\hat{P}^{(0)}_{X_1,X_2,Y} \big\| Q_{X_1,X_2,Y}\big) + \eta_1 \cdot D\big(\hat{P}^{(1)}_{X_1,Y} \big\| Q_{X_1,Y}\big) + \eta_2 \cdot D\big(\hat{P}^{(2)}_{X_2,Y} \big\| Q_{X_2,Y}\big).$$

Hence, we can rewrite the maximum likelihood solution $P^{(\mathrm{ML})}_{X_1,X_2,Y}$ as

$$P^{(\mathrm{ML})}_{X_1,X_2,Y} = \operatorname*{arg\,max}_{Q_{X_1,X_2,Y}} \mathbb{P}\{\mathscr{D}_0, \mathscr{D}_1, \mathscr{D}_2; Q_{X_1,X_2,Y}\} = \operatorname*{arg\,min}_{Q_{X_1,X_2,Y}} L^{(\mathrm{ML})}(Q_{X_1,X_2,Y}).$$

From $L(R_{X_1,X_2,Y}) = O(\epsilon^2)$, we obtain $\big\|\tilde{\ell}_{\hat{P}^{(0)}_{X_1,X_2,Y}}\big\| = O(\epsilon)$, $\big\|\tilde{\ell}_{\hat{P}^{(1)}_{X_1,Y}}\big\| = O(\epsilon)$, $\big\|\tilde{\ell}_{\hat{P}^{(2)}_{X_2,Y}}\big\| = O(\epsilon)$. By definition, we have $L(P^{(\mathrm{est})}_{X_1,X_2,Y}) < L(R_{X_1,X_2,Y}) = O(\epsilon^2)$, which implies that $\big\|\tilde{\ell}_{P^{(\mathrm{est})}_{X_1,X_2,Y}}\big\| = O(\epsilon)$.

We first consider $Q_{X_1,X_2,Y}$ with

$$\big\|\tilde{\ell}_{Q_{X_1,X_2,Y}} - \tilde{\ell}_{P^{(\mathrm{est})}_{X_1,X_2,Y}}\big\| \le \epsilon. \tag{185}$$

Then, we have

$$\big\|\tilde{\ell}_{Q_{X_1,X_2,Y}}\big\| \le \big\|\tilde{\ell}_{Q_{X_1,X_2,Y}} - \tilde{\ell}_{P^{(\mathrm{est})}_{X_1,X_2,Y}}\big\| + \big\|\tilde{\ell}_{P^{(\mathrm{est})}_{X_1,X_2,Y}}\big\| = O(\epsilon), \tag{186}$$

and it follows from Fact 7 that

$$\begin{aligned} L^{(\mathrm{ML})}(Q_{X_1,X_2,Y}) &= \eta_0 \cdot D\big(\hat{P}^{(0)}_{X_1,X_2,Y}\big\|Q_{X_1,X_2,Y}\big) + \eta_1 \cdot D\big(\hat{P}^{(1)}_{X_1,Y}\big\|Q_{X_1,Y}\big) + \eta_2 \cdot D\big(\hat{P}^{(2)}_{X_2,Y}\big\|Q_{X_2,Y}\big) \\ &= \frac{1}{2}\Big(\eta_0 \cdot \big\|\tilde{\ell}_{\hat{P}^{(0)}_{X_1,X_2,Y}} - \tilde{\ell}_{Q_{X_1,X_2,Y}}\big\|^2 + \eta_1 \cdot \big\|\tilde{\ell}_{\hat{P}^{(1)}_{X_1,Y}} - \tilde{\ell}_{Q_{X_1,Y}}\big\|^2 \\ &\qquad + \eta_2 \cdot \big\|\tilde{\ell}_{\hat{P}^{(2)}_{X_2,Y}} - \tilde{\ell}_{Q_{X_2,Y}}\big\|^2\Big) + o(\epsilon^2) \\ &= \frac{1}{2} \cdot L(Q_{X_1,X_2,Y}) + o(\epsilon^2). \end{aligned} \tag{187}$$

Therefore, for $Q_{X_1,X_2,Y}$ that satisfies (185), the minimum value of $L^{(\mathrm{ML})}(Q_{X_1,X_2,Y})$ is achieved by $Q_{X_1,X_2,Y} = P^{(\mathrm{est})}_{X_1,X_2,Y} + o(\epsilon)$.

Now we consider the case of $Q_{X_1,X_2,Y}$ with $\big\|\tilde{\ell}_{Q_{X_1,X_2,Y}} - \tilde{\ell}_{P^{(\mathrm{est})}_{X_1,X_2,Y}}\big\| > \epsilon$. Let $\epsilon' = \epsilon/\big\|\tilde{\ell}_{Q_{X_1,X_2,Y}} - \tilde{\ell}_{P^{(\mathrm{est})}_{X_1,X_2,Y}}\big\| < 1$ and define

$$\bar{Q}_{X_1,X_2,Y} \triangleq \epsilon' \cdot Q_{X_1,X_2,Y} + (1-\epsilon') \cdot P^{(\mathrm{est})}_{X_1,X_2,Y}. \tag{188}$$

Then, we have $\tilde{\ell}_{\bar{Q}_{X_1,X_2,Y}} = \epsilon' \cdot \tilde{\ell}_{Q_{X_1,X_2,Y}} + (1-\epsilon') \cdot \tilde{\ell}_{P^{(\mathrm{est})}_{X_1,X_2,Y}}$, which implies

$$\big\|\tilde{\ell}_{\bar{Q}_{X_1,X_2,Y}} - \tilde{\ell}_{P^{(\mathrm{est})}_{X_1,X_2,Y}}\big\| = \epsilon' \cdot \big\|\tilde{\ell}_{Q_{X_1,X_2,Y}} - \tilde{\ell}_{P^{(\mathrm{est})}_{X_1,X_2,Y}}\big\| = \epsilon. \tag{189}$$

As a result, we can apply the same analysis on $\bar{Q}_{X_1,X_2,Y}$ and obtain [cf. (187)]

$$L^{(\mathrm{ML})}(\bar{Q}_{X_1,X_2,Y}) = \frac{1}{2} \cdot L(\bar{Q}_{X_1,X_2,Y}) + o(\epsilon^2). \tag{190}$$

Hence, we obtain from (189) that

$$L^{(\mathrm{ML})}(\bar{Q}_{X_1,X_2,Y}) > L^{(\mathrm{ML})}(P^{(\mathrm{est})}_{X_1,X_2,Y}) = \frac{1}{2} \cdot L(P^{(\mathrm{est})}_{X_1,X_2,Y}) + o(\epsilon^2) \tag{191}$$

for ϵ sufficiently small.

In addition, since $L^{(\mathrm{ML})}$ is convex, it follows from Jensen's inequality that (cf. (188))

$$L^{(\mathrm{ML})}(\bar{Q}_{X_1,X_2,Y}) \le \epsilon' \cdot L^{(\mathrm{ML})}(Q_{X_1,X_2,Y}) + (1-\epsilon') \cdot L^{(\mathrm{ML})}(P^{(\mathrm{est})}_{X_1,X_2,Y}). \tag{192}$$

As a result, from (191) and (192) we have $L^{(\mathrm{ML})}(Q_{X_1,X_2,Y}) > L^{(\mathrm{ML})}(P^{(\mathrm{est})}_{X_1,X_2,Y})$.

Combining both cases of $Q_{X_1,X_2,Y}$, we obtain (99). ∎

C.21 Proof of Proposition 36

It suffices to establish that there exist $\alpha\colon \mathcal{U} \to \mathbb{R}$, $\beta\colon \mathcal{V} \to \mathbb{R}$, such that

$$\mathfrak{i}_{\underline{X};U}(\underline{x}, u) = \alpha(u) \cdot \left[\tanh\left(2w \cdot \varphi(\underline{x}) + b_U\right) - \tanh(b_U)\right], \tag{193a}$$

$$\mathfrak{i}_{\underline{Y};V}(\underline{y}, v) = \beta(v) \cdot \left[\tanh\left(2w \cdot \varphi(\underline{y}) + b_V\right) - \tanh(b_V)\right]. \tag{193b}$$

To see this, note that from the Markov relation $\underline{X} - U - V - \underline{Y}$, we have

$$P_{\underline{X},\underline{Y}}(\underline{x}, \underline{y}) = \sum_{u \in \mathcal{U}, v \in \mathcal{V}} P_{\underline{X}|U}(\underline{x}|u) \cdot P_{\underline{Y}|V}(\underline{y}|v) \cdot P_{U,V}(u, v)$$

$$= P_{\underline{X}}(\underline{x})P_{\underline{Y}}(\underline{y}) \sum_{u\in\mathcal{U},v\in\mathcal{V}} P_{U,V}(u,v)\cdot(1+\mathfrak{i}_{\underline{X};U}(\underline{x},u))\cdot(1+\mathfrak{i}_{\underline{Y};V}(\underline{y},v))$$

$$= P_{\underline{X}}(\underline{x})P_{\underline{Y}}(\underline{y}) \left(1+\sum_{u\in\mathcal{U},v\in\mathcal{V}} P_{U,V}(u,v)\cdot\mathfrak{i}_{\underline{X};U}(\underline{x},u)\cdot\mathfrak{i}_{\underline{Y};V}(\underline{y},v)\right),$$

where to obtain the last equality we have used the fact that

$$\sum_{u\in\mathcal{U}} P_U(u)\cdot\mathfrak{i}_{\underline{X};U}(\underline{x},u) = \sum_{v\in\mathcal{V}} P_V(v)\cdot\mathfrak{i}_{\underline{Y};V}(\underline{y},v) = 0.$$

Therefore, we obtain

$$\begin{aligned}\mathfrak{i}_{\underline{X};\underline{Y}}(\underline{x},\underline{y}) &= \frac{P_{\underline{X},\underline{Y}}(\underline{x},\underline{y})}{P_{\underline{X}}(\underline{x})P_{\underline{Y}}(\underline{y})} - 1\\ &= \sum_{u\in\mathcal{U},v\in\mathcal{V}} P_{U,V}(u,v)\cdot\mathfrak{i}_{\underline{X};U}(\underline{x},u)\cdot\mathfrak{i}_{\underline{Y};V}(\underline{y},v)\\ &= \mathbb{E}\left[\alpha(U)\beta(V)\right]\cdot\left[\tanh\left(2w\cdot\varphi(\underline{x})+b_U\right)-\tanh(b_U)\right]\cdot\left[\tanh\left(2w\cdot\varphi(\underline{y})+b_V\right)-\tanh(b_V)\right],\end{aligned}$$

which gives (105).

It remains only to establish (193). For symmetry, we consider only (193a). To begin, by definition, we have

$$P_{\underline{X}|U=u}(\underline{x}) = \frac{1}{2}\cdot\prod_{i=1}^{l-1}\left[q_u^{(1-\delta_{x_ix_{i+1}})}(1-q_u)^{\delta_{x_ix_{i+1}}}\right],$$

which implies

$$\log P_{\underline{X}|U=u}(\underline{x}) = \log\frac{1}{2} + \log q_u\cdot\sum_{i=1}^{l-1}(1-\delta_{x_ix_{i+1}}) + \log(1-q_u)\cdot\sum_{i=1}^{l-1}\delta_{x_ix_{i+1}}.$$

Therefore, we obtain

$$\begin{aligned}\frac{1}{2}\log\frac{P_{\underline{X}|U=1}(\underline{x})}{P_{\underline{X}|U=0}(\underline{x})} &= \frac{1}{2}\left[\log P_{\underline{X}|U=1}(\underline{x}) - \log P_{\underline{X}|U=0}(\underline{x})\right]\\ &= \frac{1}{2}\left[\log\frac{q_1}{q_0}\cdot\sum_{i=1}^{l-1}(1-\delta_{x_ix_{i+1}}) + \log\frac{1-q_1}{1-q_0}\cdot\sum_{i=1}^{l-1}\delta_{x_ix_{i+1}}\right]\\ &= \log\frac{q_1}{q_0}\cdot\left(\frac{l-1}{2}-\sum_{i=1}^{l-1}\delta_{x_ix_{i+1}}\right)\\ &= 2w\cdot\varphi(\underline{x}).\end{aligned}$$

As a consequence,

$$\begin{aligned}\frac{P_{\underline{X}|U=1}(\underline{x})-P_{\underline{X}|U=0}(\underline{x})}{P_{\underline{X}}(\underline{x})} &= \frac{P_{\underline{X}|U=1}(\underline{x})-P_{\underline{X}|U=0}(\underline{x})}{P_U(1)\cdot P_{\underline{X}|U=1}(\underline{x})+P_U(0)\cdot P_{\underline{X}|U=0}(\underline{x})}\\ &= \frac{1}{P_U(0)}\cdot\frac{\frac{P_{\underline{X}|U=1}(\underline{x})}{P_{\underline{X}|U=0}(\underline{x})}-1}{\frac{P_U(1)}{P_U(0)}\cdot\frac{P_{\underline{X}|U=1}(\underline{x})}{P_{\underline{X}|U=0}(\underline{x})}+1}\end{aligned}$$

$$= \frac{1}{2P_U(0)P_U(1)} \cdot \left[\frac{\frac{P_U(1)}{P_U(0)} \cdot \frac{P_{\underline{X}|U=1}(\underline{x})}{P_{\underline{X}|U=0}(\underline{x})} - 1}{\frac{P_U(1)}{P_U(0)} \cdot \frac{P_{\underline{X}|U=1}(\underline{x})}{P_{\underline{X}|U=0}(\underline{x})} + 1} - \frac{\frac{P_U(1)}{P_U(0)} - 1}{\frac{P_U(1)}{P_U(0)} + 1} \right]$$
$$= \frac{1}{2P_U(0)P_U(1)} \cdot \left[\tanh\left(2w \cdot \varphi(\underline{x}) + b_U\right) - \tanh(b_U) \right],$$

where to obtain the last equality we have used the fact that $\frac{t-1}{t+1} = \tanh\left(\frac{1}{2}\log t\right)$.

Hence, with $u' = 1 - u$, we have

$$\begin{aligned}
\mathfrak{i}_{\underline{X};U}(\underline{x}, u) &= \frac{P_{\underline{X},U}(\underline{x}, u) - P_{\underline{X}}(\underline{x})P_U(u)}{P_{\underline{X}}(\underline{x})P_U(u)} \\
&= \frac{P_{\underline{X}|U=u}(\underline{x}) - P_{\underline{X}}(\underline{x})}{P_{\underline{X}}(\underline{x})} \\
&= \frac{P_{\underline{X}|U=u}(\underline{x}) - P_U(u)P_{\underline{X}|U=u}(\underline{x}) - P_U(u')P_{\underline{X}|U=u'}(\underline{x})}{P_{\underline{X}}(\underline{x})} \\
&= P_U(u') \cdot \frac{P_{\underline{X}|U=u}(\underline{x}) - P_{\underline{X}|U=u'}(\underline{x})}{P_{\underline{X}}(\underline{x})} \\
&= P_U(u') \cdot (-1)^{u+1} \cdot \frac{P_{\underline{X}|U=1}(\underline{x}) - P_{\underline{X}|U=0}(\underline{x})}{P_{\underline{X}}(\underline{x})} \\
&= \frac{(-1)^{u+1}}{2P_U(u)} \cdot \left[\tanh\left(2w \cdot \varphi(\underline{x}) + b_U\right) - \tanh(b_U) \right],
\end{aligned}$$

which gives (193a) as desired, with $\alpha(u) = \frac{(-1)^{u+1}}{2P_U(u)}$. ■

Appendix D. Implementation Details of Experiments

We implement our experiments in Python 3 (Van Rossum and Drake, 2009), where we use the `PyTorch` (Paszke et al., 2019) library for neural network training and use the `Matplotlib` (Hunter, 2007) library for plotting. We also make use of `NumPy` (Harris et al., 2020) and `SciPy` (Virtanen et al., 2020) for the computation.

In the experiments, we apply Adam (Kingma and Ba, 2015) as the optimizer with the default parameters: a learning rate of 10^{-3}, $\beta_1 = 0.9, \beta_2 = 0.999$, and $\epsilon = 10^{-8}$. For each MLP (multilayer perceptron) used in the experiments, we set the activation function to be the softplus function $x \mapsto \log(1 + e^x)$, which are applied to all layers except the output layer.

It is worth mentioning that our choices of network architectures, optimizers and hyperparameters are not optimized with respect to the used data distributions. It is possible to further optimize such choices to improve the performance or convergence.

D.1 Learning Maximal Correlation Functions

We first introduce the implementation details for Section 7.1, where the goal is to learn maximal correlation functions for different data. The corresponding learning objective is the nested H-score (38), which are maximized during the training.

D.1.1 Implementation of Section 7.1.1

We set $|\mathcal{X}| = 8, |\mathcal{Y}| = 6$, and feature dimension $k = 3$. To generate the discrete distributions $P_{X,Y}$, we draw $(|\mathcal{X}| \cdot |\mathcal{Y}|)$ i.i.d. numbers from Unif$[0, 1]$ and divide each number by their sum. We then

use the resulting $|\mathcal{X}| \times |\mathcal{Y}|$ table as the values for the probability mass function $P_{X,Y}$. To ensure reproducible results, we set the random seed of `NumPy` to 20 230 606 in the generating process.

Then, we generate $N = 30\,000$ training sample pairs of (X, Y) from $P_{X,Y}$, then apply one-hot encoding such that the inputs are represented as $|\mathcal{X}|$ and $|\mathcal{Y}|$ dimensional vectors. Then, we use two one-layer linear networks as the feature extractors f and g.

We train the networks with a minibatch size of 128 for 100 epochs. Then, we obtain the estimated f_i^*, g_i^*, and σ_i by applying (40) and compare them with corresponding theoretical values, which we compute from the SVD of corresponding CDM matrix [cf. (120)], with the results shown in Figure 14. Note that since $f_i^* \otimes g_i^* = (-f_i^*) \otimes (-g_i^*)$, both (f_i^*, g_i^*) and $(-f_i^*, -g_i^*)$ are the optimal feature pairs. For the sake of presentation, we applied a sign modification before the plotting.

D.1.2 Implementation of Section 7.1.2

In this experiment, we first generate $N = 50\,000$ samples of $(X, Y) \in \mathbb{R}^2$ for training, to learn $k = 2$ dimensional features f and g. We use two MLPs of the same architecture as the feature extractors for f and g. Specifically, each MLP is with three layers, where the dimensions for all intermediate features, from input to output, are: input = 1 – 32 – 32 – 2 = output. We then train the networks with a minibatch size of 256 for 100 epochs and use the learned features for estimation tasks, as demonstrated in Section 7.1.2.

D.1.3 Implementation of Section 7.1.3

In this experiment, we set $k = 1$. To extract f and g from input sequences $\underline{X}$ and $\underline{Y}$, we use one-dimensional convolutional neural networks as the feature extractors, which are used in sentence classifications (Kim, 2014; Zhang and Wallace, 2017). In particular, f and g are of the same architecture, composed of an embedding (linear) layer, a 1 dimensional convolutional layer, an average pooling layer, and a fully connected (linear) layer.

We use feature extractor f as an example to illustrate the processing of sequential data. First, we represent $\underline{x}$ sequence as a one-hot encoded list, i.e., each $x_i \in \{(1,0)^{\mathrm{T}}, (0,1)^{\mathrm{T}}\}$. Then, the embedding layer maps each x_i to a 4-dimensional vector. The one-dimensional convolutional layer then processes the length-l list of embedded 4-dimensional vectors, by 32 convolutional kernels of size 4. We then activate the convolution results by the ReLU function $x \mapsto \max\{x, 0\}$. The output from each convolutional kernel is further averaged by the average pooling layer, leading to a 32 dimensional feature, with each dimension corresponding to a convolutional kernel. Finally, we feed the 32 dimensional feature to the fully connected layer and generate $k = 1$ dimensional output.

Then, we train the feature extractors f and g with a minibatch size of 128 for 100 epochs. The learned features are shown in Section 7.1.3.

D.2 Learning With Orthogonality Constraints

In this experiment, we use the same dataset generated in Appendix D.1.2. We set $\bar{k} = k = 1$, i.e., we learn one-dimensional feature f from X orthogonal to given one-dimensional $\bar{f}$. To this end, we use three MLPs of the same architecture as the feature extractors for $\bar{g}, f, g$, with dimensions input = 1 – 32 – 32 – 1 = output. We then train the networks with a minibatch size of 256 for 100 epochs to maximize the nested H-score restricted to $\bar{f} = \phi$ [cf. (52)], for $\phi(x) = x$ and $\phi(x) = x^2$, respectively.

D.3 Learning With Side Information

We set $|\mathcal{X}| = 8, |\mathcal{S}| = |\mathcal{Y}| = 3$, and generate $P_{X,S,Y}$ in a manner similar to Appendix D.1.1, with the same random seed. Then, we generate $N = 50\,000$ training samples of (X, S, Y) triples. In our implementation, we set $\bar{k} = |\mathcal{S}| - 1 = 2$ and $k = 1$, and set feature extractors $\bar{f} \in \mathcal{F}_{\mathcal{X}}^2, \bar{g} \in \mathcal{F}_{\mathcal{S}}^2$, $f \in \mathcal{F}_{\mathcal{X}}$, $g \in \mathcal{F}_{\mathcal{S}\times\mathcal{Y}}$ as corresponding one-layer linear networks with one-hot encoded inputs. In particular, we convert each (s, y) to one unique one-hot vector of in $\mathbb{R}^{|\mathcal{S}|\cdot|\mathcal{Y}|}$ as the input to the network g. Then, we train these feature extractors on the training set with a minibatch size of 256 for 100 epochs, to maximize the nested H-score configured by $\mathcal{C}_{\mathsf{MC}}$ [cf. (59)].

For comparison, we train a multihead network shown in Figure 11 on the same dataset, with the same minibatch size and epochs. The feature f is again implemented by a one-layer linear network. In particular, we maximize the log-likelihood function (70) to learn the corresponding feature and weights. Then, we convert the weights to $g \in \mathcal{F}_{\mathcal{S}\times\mathcal{Y}}$ via the correspondence [cf. (69)] $g(s, y) = G_s(1, y)$. The comparison between features learned from two approaches are shown in Figure 20. For the sake of presentation, we have normalized the features before plotting, such that f and each $g(s, \cdot)$ are zero-mean, and unit variance with respect to $\mathrm{Unif}(\mathcal{X})$ and $\mathrm{Unif}(\mathcal{Y})$, respectively.

D.4 Multimodal Learning With Missing Modalities

D.4.1 Implementation of Section 7.4.1

We first generate $N = 50\,000$ triples of (X_1, X_2, Y) for training. In implementing the algorithm, we $\bar{k} = k = 1$. To represent $\bar{f} \in \mathcal{F}_{\mathcal{X}}$, we set each $\bar{f}^{(i)}$, $i \in \{1, 2\}$ as an MLP with dimensions input = 1 – 32 – 32 – 1 = output. To represent f, we use an MLP with dimensions input = 2 – 32 – 32 – 1 = output with the input set to $X_1 + X_2 = (X_1, X_2)^{\mathrm{T}}$. Since Y is discrete, we use one-layer linear network as $\bar{g}$ and g, where the inputs are one-hot encoded Y. Therefore, both $\bar{g}$ and g are with one linear layer, taking $|\mathcal{Y}| = 2$ dimensional input and outputting one-dimensional feature.

We then train these feature extractors on the training set with a minibatch size of 256 for 100 epochs, to maximize the nested H-score configured by $\mathcal{C}_{\mathsf{BI}}$ [cf. (78)]. The learned features are then shown in Figure 21.

For the prediction problem with missing modality, we use two MLPs to represent conditional expectation operators $\phi_1 \triangleq \tau_1(\bar{f}^{(2)})$ and $\phi_2 \triangleq \tau_2(\bar{f}^{(1)})$, each with dimensions input = 1 – 32 – 32 – 1 = output. These two networks are optimized by minimizing the corresponding mean square error. For example, we train ϕ_1 by minimizing $\mathbb{E}\left[\left(\bar{f}^{(2)}(X_2) - \phi_1(X_1)\right)^2\right]$ over all parameters in ϕ_1 network. The training of ϕ_1, ϕ_2 is with minibatch size of 256, and 100 epochs.

D.4.2 Implementation of Section 7.4.2

We generate $N = 50\,000$ triples of (X_1, X_2, Y) from (106) and (111), and adopt the decomposition (89) on each triple. This gives three pairwise datasets with samples of (X_1, X_2), (X_1, Y), (X_2, Y), where each dataset has N sample pairs. Then, we adopt the same setting of networks to represent one-dimensional $\bar{f}^{(1)}, \bar{f}^{(2)}$ and g.

We then train these feature extractors on the three datasets for 100 epochs with a minibatch size of 256, to maximize $\mathscr{H}(\bar{f}, \bar{g})$. Here, we compute $\mathscr{H}(\bar{f}, \bar{g})$ based on the minibatches from the three pairwise datasets according to (93).

To solve the prediction problem with missing modality, we use the same network architectures and training settings to learn $\tau_1(\bar{f}^{(2)})$ and $\tau_2(\bar{f}^{(1)})$, as introduced in Appendix D.4.1.

References

S-I Amari. Information geometry on hierarchy of probability distributions. *IEEE transactions on information theory*, 47(5):1701–1711, 2001.

Shun-ichi Amari and Hiroshi Nagaoka. *Methods of information geometry*, volume 191. American Mathematical Soc., 2000.

Galen Andrew, Raman Arora, Jeff Bilmes, and Karen Livescu. Deep canonical correlation analysis. In *International conference on machine learning*, pages 1247–1255. PMLR, 2013.

Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeshwar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and Devon Hjelm. Mutual information neural estimation. In *International conference on machine learning*, pages 531–540. PMLR, 2018.

Leo Breiman and Jerome H Friedman. Estimating optimal transformations for multiple regression and correlation. *Journal of the American statistical Association*, 80(391):580–598, 1985.

Michael M Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. Geometric deep learning: going beyond euclidean data. *IEEE Signal Processing Magazine*, 34(4): 18–42, 2017.

Michael M Bronstein, Joan Bruna, Taco Cohen, and Petar Veličković. Geometric deep learning: Grids, groups, graphs, geodesics, and gauges. *arXiv preprint arXiv:2104.13478*, 2021.

Andreas Buja. Remarks on functional canonical variates, alternating least squares methods and ace. *The Annals of Statistics*, pages 1032–1069, 1990.

R Caruana. Multitask learning: A knowledge-based source of inductive bias1. In *Proceedings of the Tenth International Conference on Machine Learning*, pages 41–48. Citeseer, 1993.

Kwan Ho Ryan Chan, Yaodong Yu, Chong You, Haozhi Qi, John Wright, and Yi Ma. Redunet: A white-box deep network from the principle of maximizing rate reduction. *The Journal of Machine Learning Research*, 23(1):4907–5009, 2022.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

Yuxin Chen, Yuejie Chi, Jianqing Fan, Cong Ma, et al. Spectral methods for data science: A statistical perspective. *Foundations and Trends® in Machine Learning*, 14(5):566–806, 2021.

Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

Thomas M Cover and Joy A Thomas. Elements of information theory (wiley series in telecommunications and signal processing), 2006.

George Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems*, 2(4):303–314, 1989.

Flavio du Pin Calmon, Ali Makhdoumi, Muriel Médard, Mayank Varia, Mark Christiansen, and Ken R Duffy. Principal inertia components and applications. *IEEE Transactions on Information Theory*, 63(8):5011–5038, 2017.

Nelson Dunford and Jacob T Schwartz. *Linear operators, part 1: general theory*, volume 10. John Wiley & Sons, 1988.

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

Artur Ekert and Peter L Knight. Entangled quantum systems and the schmidt decomposition. *American Journal of Physics*, 63(5):415–423, 1995.

Ronald A Fisher. The use of multiple measurements in taxonomic problems. *Annals of eugenics*, 7(2):179–188, 1936.

Hans Gebelein. Das statistische problem der korrelation als variations-und eigenwertproblem und sein zusammenhang mit der ausgleichsrechnung. *ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik*, 21(6):364–379, 1941.

Michael Greenacre. *Correspondence analysis in practice*. CRC press, 2017.

Jeff Z HaoChen, Colin Wei, Adrien Gaidon, and Tengyu Ma. Provable guarantees for self-supervised deep learning with spectral contrastive loss. *Advances in Neural Information Processing Systems*, 34:5000–5011, 2021.

Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. Array programming with NumPy. *Nature*, 585(7825):357–362, September 2020. doi: 10.1038/s41586-020-2649-2. URL `https://doi.org/10.1038/s41586-020-2649-2`.

Trevor Hastie, Robert Tibshirani, Jerome H Friedman, and Jerome H Friedman. *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer, 2009.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

Hermann O Hirschfeld. A connection between correlation and contingency. In *Proceedings of the Cambridge Philosophical Society*, volume 31, pages 520–524, 1935.

Thomas Hofmann, Bernhard Schölkopf, and Alexander J Smola. Kernel methods in machine learning. 2008.

Harold Hotelling. Relations between two sets of variates. *Biometrika*, 28(3/4):321–377, 1936.

Hsiang Hsu, Salman Salamatian, and Flavio P Calmon. Generalizing correspondence analysis for applications in machine learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(12):9347–9362, 2021.

Shao-Lun Huang and Xiangxiang Xu. On the sample complexity of HGR maximal correlation functions for large datasets. *IEEE Transactions on Information Theory*, 67(3):1951–1980, 2020.

Shao-Lun Huang, Anuran Makur, Gregory W. Wornell, and Lizhong Zheng. Universal features for high-dimensional learning and inference. *Foundations and Trends® in Communications and Information Theory*, 21(1-2):1–299, 2024. ISSN 1567-2190. doi: 10.1561/0100000107. URL http://dx.doi.org/10.1561/0100000107.

John D Hunter. Matplotlib: A 2D graphics environment. *Computing in science & engineering*, 9 (03):90–95, 2007.

Edwin T Jaynes. Information theory and statistical mechanics. *Physical review*, 106(4):620, 1957a.

Edwin T Jaynes. Information theory and statistical mechanics. ii. *Physical review*, 108(2):171, 1957b.

Yoon Kim. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar, October 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1181. URL https://aclanthology.org/D14-1181.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1412.6980.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017.

Henry Oliver Lancaster. The structure of bivariate distributions. *The Annals of Mathematical Statistics*, 29(3):719–736, 1958.

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

Anuran Makur, Gregory W Wornell, and Lizhong Zheng. On estimation of modal decompositions. In *2020 IEEE International Symposium on Information Theory (ISIT)*, pages 2717–2722. IEEE, 2020.

F Gustav Mehler. Ueber die entwicklung einer function von beliebig vielen variablen nach laplaceschen functionen höherer ordnung. 1866.

Jiquan Ngiam, Aditya Khosla, Mingyu Kim, Juhan Nam, Honglak Lee, and Andrew Y Ng. Multimodal deep learning. In *ICML*, 2011.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

Karl Pearson. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin philosophical magazine and journal of science*, 2(11):559–572, 1901.

Karl Pearson. Mathematical contributions to the theory of evolution XIII: On the theory of contingency and its relation to association and normal correlation. Draper's Co. Research Memoirs. Biometric Series, No. 1 (Reprinted 1948), 1904.

Alfréd Rényi. On measures of dependence. *Acta Mathematica Academiae Scientiarum Hungarica*, 10(3-4):441–451, 1959.

Sebastian Ruder. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*, 2017.

Igal Sason and Sergio Verdú. f-divergence inequalities. *IEEE Transactions on Information Theory*, 62(11):5973–6006, 2016. doi: 10.1109/TIT.2016.2603151.

Erhard Schmidt. Zur theorie der linearen und nichtlinearen integralgleichungen. i. teil: Entwicklung willkürlicher funktionen nach systemen vorgeschriebener. *Mathematische Annalen*, 63:433–476, 1907.

Claude E Shannon. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423, 1948.

Naftali Tishby and Noga Zaslavsky. Deep learning and the information bottleneck principle. In *2015 ieee information theory workshop (itw)*, pages 1–5. IEEE, 2015.

Naftali Tishby, Fernando C Pereira, and William Bialek. The information bottleneck method. *arXiv preprint physics/0004057*, 2000.

Guido Van Rossum and Fred L. Drake. *Python 3 Reference Manual*. CreateSpace, Scotts Valley, CA, 2009. ISBN 1441412697.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Pauli Virtanen, Ralf Gommers, Travis E Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, et al. Scipy 1.0: fundamental algorithms for scientific computing in python. *Nature methods*, 17(3):261–272, 2020.

Lichen Wang, Jiaxiang Wu, Shao-Lun Huang, Lizhong Zheng, Xiangxiang Xu, Lin Zhang, and Junzhou Huang. An efficient approach to informative feature extraction from multimodal data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 5281–5288, 2019.

Joachim Weidmann. *Linear operators in Hilbert spaces*, volume 68. Springer Science & Business Media, 2012.

Xiangxiang Xu and Shao-Lun Huang. Maximal correlation regression. *IEEE Access*, 8:26591–26601, 2020.

Xiangxiang Xu and Shao-Lun Huang. An information theoretic framework for distributed learning algorithms. In *2021 IEEE International Symposium on Information Theory (ISIT)*, pages 314–319. IEEE, 2021.

Xiangxiang Xu and Lizhong Zheng. Multivariate feature extraction. In *2022 58th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 1–8. IEEE, 2022.

Xiangxiang Xu and Lizhong Zheng. Kernel subspace and feature extraction. In *2023 IEEE International Symposium on Information Theory (ISIT) (ISIT'2023)*, pages 1032–1037, Taipei, Taiwan, June 2023a.

Xiangxiang Xu and Lizhong Zheng. Sequential dependence decomposition and feature learning. In *2023 59th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 1–8. IEEE, 2023b.

Xiangxiang Xu, Shao-Lun Huang, Lizhong Zheng, and Lin Zhang. The geometric structure of generalized softmax learning. In *2018 IEEE Information Theory Workshop (ITW)*, pages 1–5. IEEE, 2018.

Xiangxiang Xu, Shao-Lun Huang, Lizhong Zheng, and Gregory W Wornell. An information theoretic interpretation to deep neural networks. *Entropy*, 24(1):135, 2022.

Nicholas Young. *An introduction to Hilbert space*. Cambridge university press, 1988.

Ye Zhang and Byron C Wallace. A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 253–263, 2017.