entropy

MDPI

Article

An Information Theoretic Interpretation to Deep Neural Networks †

Xiangxiang Xu [1], Shao-Lun Huang [1,*], Lizhong Zheng [2] and Gregory W. Wornell [2]

[1] Data Science and Information Technology Research Center, Tsinghua–Berkeley Shenzhen Institute, Shenzhen 518055, China; xuxx@mit.edu
[2] Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA 02139, USA; lizhong@mit.edu (L.Z.); gww@mit.edu (G.W.W.)
* Correspondence: shaolun.huang@sz.tsinghua.edu.cn
† This work was presented in part at the 2019 IEEE International Symposium on Information Theory (ISIT), Paris, France, 7–12 July 2019.

Abstract: With the unprecedented performance achieved by deep learning, it is commonly believed that deep neural networks (DNNs) attempt to extract informative features for learning tasks. To formalize this intuition, we apply the local information geometric analysis and establish an information-theoretic framework for feature selection, which demonstrates the information-theoretic optimality of DNN features. Moreover, we conduct a quantitative analysis to characterize the impact of network structure on the feature extraction process of DNNs. Our investigation naturally leads to a performance metric for evaluating the effectiveness of extracted features, called the H-score, which illustrates the connection between the practical training process of DNNs and the information-theoretic framework. Finally, we validate our theoretical results by experimental designs on synthesized data and the ImageNet dataset.

Keywords: deep neural network; information theory; local information geometry; feature extraction

Citation: Xu, X.; Huang, S.-L.; Zheng, L.; Wornell, G.W. An Information Theoretic Interpretation to Deep Neural Networks. *Entropy* **2022**, *24*, 135. https://doi.org/10.3390/e24010135

Academic Editor: Raúl Alcaraz

Received: 7 December 2021
Accepted: 12 January 2022
Published: 17 January 2022

1. Introduction

Due to the striking performance of deep learning in various application fields, deep neural networks (DNNs) have gained great attention in modern computer science. While it is a common understanding that the features extracted from the hidden layers of DNN are "informative" for learning tasks, the mathematical meaning of informative features in DNN is generally not clear. From the practical perspective, DNN models have obtained unprecedented performance in varying tasks, such as image recognition [1], language processing [2,3], and games [4,5]. However, the understanding of the feature extraction behind these models is relatively lacking, which poses challenges for their application in security-sensitive tasks, such as the autonomous vehicle.

To address this problem, there have been numerous research efforts, including both experimental and theoretical studies [6]. The experimental studies usually focus on some empirical properties of the feature extracted by DNNs, by visualizing the feature [7] or testing its performance on specific training settings [8] or learning tasks [9]. Though such empirical methods have provided some intuitive interpretations, the performance can highly depend on the data and network architecture used. For example, while the feature visualization works well on convolutional neural networks, its application to other networks is typically less effective [10].

In contrast, theoretical studies focus on the analytical properties of the extracted feature or the learning process in DNNs. Due to the complicated structure of DNNs, existing studies were often restricted to the networks of specific structures, e.g., network with infinite width [11] or two-layer network [12,13], to characterize the theoretical behaviors. However, the interpretation of the optimal feature remains unclear, which limits their

further applications. To obtain better interpretability, tools and measures from information theory [14] have recently been applied to connect DNNs with general information processing problems [15]. For instance, the information bottleneck [16,17] employs the mutual information as the metric to quantify the informativeness of features in DNN, and other information metrics, such as the Kullback–Leibler (KL) divergence [18] and Weissenstein distance [19], are also used in different problems. However, there is still a disconnection between these information metrics and the performance objectives of the inference tasks that DNNs want to solve [20]. Therefore, it is, in general, difficult to match the DNN learning with the optimization of a particular information metric.

This paper aims to provide an information-theoretic interpretation to the feature extraction process in DNNs, to bridge the gap between the practical deep learning implementations and information-theoretic characterizations. To this end, we first propose an information-theoretic feature selection framework, which establishes an information metric to measure the performance of each given feature in inference tasks. In addition, we demonstrate that the optimal features extracted by DNNs coincide with the solutions of the information-theoretic feature selection problem, which share the same performance metric. Therefore, our results give an explicit interpretation of the learning goal of the back-propagation (BackProp) and stochastic gradient descent (SGD) operations in deep learning [21], which also lead to a performance metric for evaluating the effectiveness of the extracted features. Finally, we validate our theoretic characterizations using numerical experiments on both synthesized data and the ImageNet [22] dataset for image classification.

2. Preliminaries and Methods

2.1. Methodological Background

The main method used in our development is local information geometry [23,24], which characterizes the local geometric properties of the probability distribution space. The local information geometric method is closely related to the conventional Hirschfeld–Gebelein–Rényi (HGR) maximal correlation [25–27] problem, which has attracted increasing interest in the information theory community [28–33], and has also been applied in data analysis [34] and privacy studies [35].

Specifically, we use the local information geometric method to construct and investigate an information-theoretic feature selection problem in Section 3.1, which leads to an information metric of features and also demonstrates an SVD (singular value decomposition) structure of the feature selection process. Following the same analysis framework, we characterize the optimal feature extracted by DNNs in Section 3.2, and demonstrate that the same SVD structure is shared by DNNs. Based on the established connection, we then propose an effectiveness measure for DNNs, with details presented in Section 3.3.

2.2. Notations

Throughout this paper, we use X, $\mathcal{X}$, P_X, and x to represent a discrete random variable, the range, the probability distribution, and the value of X. In addition, for any function $s(X) \in \mathbb{R}^k$ of X, we use μ_s to denote the mean of $s(X)$, and "~" to denote the centered variable with mean subtracted, e.g., $\tilde{s}(X) \triangleq s(X) - \mu_s$. Moreover, we use $\|\cdot\|$ and $\|\cdot\|_{\mathrm{F}}$ to denote the ℓ_2-norm and the Frobenius norm, respectively. All logarithms in our analyses are base e, i.e., natural.

2.3. Local Information Geometry

The following concepts from local information geometry would be useful in our development.

Definition 1 (ϵ-Neighborhood). *Let $\mathcal{P}^{\mathcal{X}}$ denote the space of distributions on some finite alphabet $\mathcal{X}$, and let* relint$(\mathcal{P}^{\mathcal{X}})$ *denote the subset of strictly positive distributions. For a given $\epsilon > 0$, the ϵ-neighborhood of a distribution $P_X \in$* relint$(\mathcal{P}^{\mathcal{X}})$ *is defined by the χ^2-divergence as*

$$\mathcal{N}_\epsilon^{\mathcal{X}}(P_X) \triangleq \left\{ P \in \mathcal{P}^{\mathcal{X}} \colon \sum_{x \in \mathcal{X}} \frac{\big(P(x) - P_X(x)\big)^2}{P_X(x)} \leq \epsilon^2 \right\}.$$

Definition 2 (ϵ-Dependence). *The random variables X, Y are called ϵ-dependent if $P_{XY} \in \mathcal{N}_\epsilon^{\mathcal{X}\times\mathcal{Y}}(P_X P_Y)$.*

Definition 3 (ϵ-Attribute). *A random variable U is called an ϵ-attribute of X if $P_{X|U}(\cdot|u) \in \mathcal{N}_\epsilon^{\mathcal{X}}(P_X)$, for all $u \in \mathcal{U}$.*

We will focus on the small ϵ regime, which we refer to as the *local analysis regime*. In addition, for any $P \in \mathcal{P}^{\mathcal{X}}$, we define the *information vector* ϕ and *feature function* $L(x)$ corresponding to P, with respect to a reference distribution $P_X \in$ relint$(\mathcal{P}^{\mathcal{X}})$, as

$$\phi(x) \triangleq \frac{P(x) - P_X(x)}{\sqrt{P_X(x)}}, \quad L(x) \triangleq \frac{\phi(x)}{\sqrt{P_X(x)}}. \tag{1}$$

This gives a three way correspondence $P \leftrightarrow \phi \leftrightarrow L$ for all distributions in $\mathcal{N}_\epsilon^{\mathcal{X}}(P_X)$, which will be useful in our derivations.

2.4. Modal Decomposition

Given a pair of discrete random variables X, Y with the joint distribution $P_{XY}(x, y)$, the $|\mathcal{Y}| \times |\mathcal{X}|$ matrix $\tilde{\mathbf{B}}$ is defined as

$$\tilde{\mathbf{B}}(y, x) \triangleq \frac{P_{XY}(x, y) - P_X(x)P_Y(y)}{\sqrt{P_X(x)P_Y(y)}}, \tag{2}$$

where $\tilde{\mathbf{B}}(y, x)$ is the (y, x)th entry of $\tilde{\mathbf{B}}$. The matrix $\tilde{\mathbf{B}}$ is referred to as the canonical dependence matrix (CDM) [24]. The SVD of $\tilde{\mathbf{B}}$ is referred to as the *modal decomposition* [24] of the joint distribution P_{XY}, which has the following property [18].

Lemma 1. *The SVD of $\tilde{\mathbf{B}}$ can be written as $\tilde{\mathbf{B}} = \sum_{i=1}^{K} \sigma_i \boldsymbol{\psi}_i^Y (\boldsymbol{\psi}_i^X)^{\mathrm{T}}$, where $K \triangleq \min\{|\mathcal{X}|, |\mathcal{Y}|\}$, and σ_i denotes the ith singular value with the ordering $1 \geq \sigma_1 \geq \cdots \geq \sigma_K = 0$, and $\boldsymbol{\psi}_i^Y$ and $\boldsymbol{\psi}_i^X$ are the corresponding left and right singular vectors with $\psi_K^X(x) = \sqrt{P_X(x)}$ and $\psi_K^Y(y) = \sqrt{P_Y(y)}$.*

This SVD decomposes the feature spaces of X, Y into maximally correlated features. To see that, consider the generalized canonical correlation analysis (CCA) problem:

$$\max_{\substack{\mathbb{E}[f_i(X)]=\mathbb{E}[g_i(Y)]=0 \\ \mathbb{E}[f_i(X)\, f_j(X)]=\mathbb{E}[g_i(Y)\, g_j(Y)]=\delta_{ij}}} \sum_{i=1}^{k} \mathbb{E}[f_i(X)\, g_i(Y)], \tag{3}$$

where δ_{ij} denotes the Kronecker delta function. It can be shown that for any $1 \leq k \leq K-1$, the optimal features are $f_i(x) = \psi_i^X(x)/\sqrt{P_X(x)}$, and $g_i(y) = \psi_i^Y(y)/\sqrt{P_Y(y)}$, for $i = 0, \ldots, K-1$, where $\psi_i^X(x)$ and $\psi_i^Y(y)$ are the xth and yth entries of $\boldsymbol{\psi}_i^X$ and $\boldsymbol{\psi}_i^Y$, respectively [18]. The special case $k = 1$ corresponds to the HGR maximal correlation [25–27], and the optimal features can be computed from the ACE (Alternating Conditional Expectation) algorithm [36].

2.5. Deep Neural Networks

The architecture of deep neural networks (under log-loss) can be depicted as Figure 1, where X is the input data, e.g., images, audios, or natural languages. Moreover, Y is the objective to predict, which can represent a discrete label in classification tasks, or represent target natural languages in machine translations [37]. Specifically, for given data X, the network produces a (trainable) feature mapping to generate k-dimensional feature $s(x) = (s_1, \ldots, s_k)^{\mathrm{T}}$. In practice, the feature mapping block (depicted as the gray block in Figure 1) is typically composed of hundreds and thousands of functional components (e.g., residual block [1]) with different types of layers, and may contain recurrent structure, e.g., LSTM (Long Short-Term Memory) [38]. In general, the internal structure of the feature mapping can have various different types of designs, depending on the learning tasks.

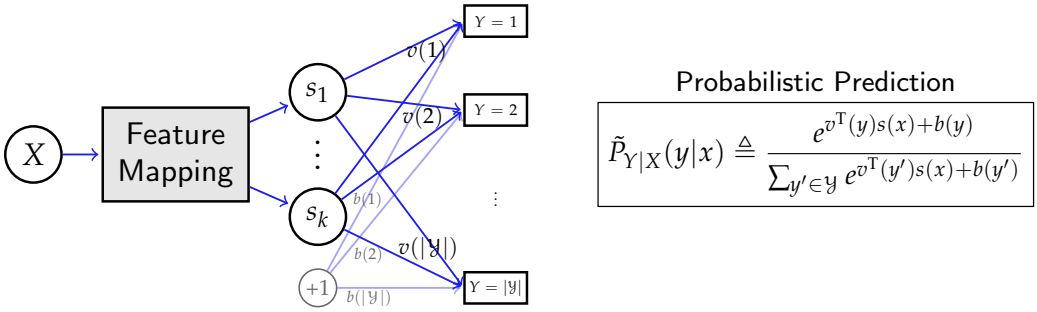

Figure 1. A deep neural network that uses data X to predict Y. All hidden layers together map the input data X to k-dimensional feature $s(x) = (s_1, \ldots, s_k)^{\mathrm{T}}$. Then, the probabilistic prediction $\tilde{P}_{Y|X}$ of Y is computed from $s(x), v(y)$, and $b(y)$, where v and bias b are the weights and bias in the last layer.

After obtaining the feature $s(X)$, the Y is then predicted by the probability distribution $\tilde{P}^{(s,v,b)}_{Y|X}$ of the form

$$\tilde{P}^{(s,v,b)}_{Y|X}(y|x) \triangleq \frac{e^{v^{\mathrm{T}}(y)s(x)+b(y)}}{\sum_{y' \in \mathcal{Y}} e^{v^{\mathrm{T}}(y')s(x)+b(y')}}, \tag{4}$$

which is obtained by applying the softmax function [39] on $v^{\mathrm{T}}(y)s(x) + b(y)$, where $v(\cdot)$ and $b(\cdot)$ are the weights and biases in the last layer, respectively (this is equivalent to the common practice that denotes weight and biases by the matrix $[v(1), \ldots, v(|\mathcal{Y}|)]^{\mathrm{T}}$ and the vector $[b(1), \ldots, b(|\mathcal{Y}|)]^{\mathrm{T}}$, respectively. However, as we will show later, expressing weights v and biases b as mappings of y can better illustrate their roles in feature selection). We will use $\tilde{P}_{Y|X}$ to refer to $\tilde{P}^{(s,v,b)}_{Y|X}$ when there is no ambiguity.

Then, for a given training set of labeled samples (x_i, y_i), for $i = 1, \ldots, N$, all the parameters in the network, including v, b, as well as those in the feature mapping block, are chosen to maximize the log-likelihood function (or, equivalently, minimize the log-loss)

$$\frac{1}{N} \sum_{i=1}^{N} \log \tilde{P}_{Y|X}(y_i|x_i). \tag{5}$$

The procedure of choosing such parameters is called the training of network, which can be performed by stochastic gradient descent (SGD) or its variants [21]. With a trained network, the label $\hat{y}$ for a new data sample x can be predicted by the maximum a posteriori (MAP) estimation, i.e., $\hat{y} = \arg\max_{y \in \mathcal{Y}} \tilde{P}_{Y|X}(y|x)$. Specifically, when we make predictions for samples in a test dataset, the proportion of samples with correct prediction (i.e., $\hat{y} = y$) over all samples is called the test accuracy.

3. Results

3.1. Information-Theoretic Feature Selection

Suppose that, given random variables X, Y with joint distribution P_{XY}, we want to infer about an attribute V of Y from observed i.i.d. samples $x_1, \ldots, x_n$ of X. When the statistical model $P_{X|V}$ is known, the optimal decision rule is the log-likelihood ratio test, where the log-likelihood function can be viewed as the optimal feature for inference. However, in many practical situations [18], it is hard to identify the model of the targeted attribute, and it is necessary to select low-dimensional informative features of X for inference tasks before knowing the model. An information-theoretic formulation of such feature selection problem is the universal feature selection problem [24], which we formalize as follows.

To begin, for an attribute V, we refer to $\mathcal{C}_\mathcal{Y} = \{\mathcal{V}, \{P_V(v), v \in \mathcal{V}\}, \{\boldsymbol{\phi}_v^{Y|V}, v \in \mathcal{V}\}\}$, as the *configuration* of V, where $\phi_v^{Y|V} \leftrightarrow P_{Y|V}(\cdot|v)$ is the information vector specifying the corresponding conditional distribution $P_{Y|V}(\cdot|v)$. The configuration of V models the statistical correlation between V and Y. In the sequel, we focus on the local analysis regime, for which we assume that all the attributes V of our interests to detect are ϵ-attributes of Y. As a result, the corresponding configuration satisfies $\|\boldsymbol{\phi}_v^{Y|V}\| \leq \epsilon$, for all $v \in \mathcal{V}$. We refer to such configurations as *ϵ-configurations*. The configuration of V is unknown in advance but assumed to be generated from a *rotational invariant ensemble (RIE)*.

Definition 4 (RIE). *Two configurations $\mathcal{C}_\mathcal{Y}$ and $\tilde{\mathcal{C}}_\mathcal{Y}$ defined as*

$$\mathcal{C}_\mathcal{Y} \triangleq \{\mathcal{V}, \{P_V(v), v \in \mathcal{V}\}, \{\boldsymbol{\phi}_v^{Y|V}, v \in \mathcal{V}\}\},$$

$$\tilde{\mathcal{C}}_\mathcal{Y} \triangleq \{\mathcal{V}, \{P_V(v), v \in \mathcal{V}\}, \{\tilde{\boldsymbol{\phi}}_v^{Y|V}, v \in \mathcal{V}\}\}$$

are called rotationally equivalent, if there exists a unitary matrix $\mathbf{Q}$ such that $\tilde{\boldsymbol{\phi}}_v^{Y|V} = \mathbf{Q}\,\boldsymbol{\phi}_v^{Y|V}$, for all $v \in \mathcal{V}$. Moreover, a probability measure defined on a set of configurations is called an RIE, if all rotationally equivalent configurations have the same measure.

The RIE can be interpreted as assigning a uniform measure to the attributes with the same level of distinguishability. To infer about the attribute V, we construct a k-dimensional feature vector $h^k = (h_1, \ldots, h_k)$, for some $1 \leq k \leq K-1$, of the form

$$h_i = \frac{1}{n}\sum_{l=1}^{n} f_i(x_l), \quad i = 1, \ldots, k, \tag{6}$$

for some choices of feature functions f_i. Our goal is to determine the f_i such that the optimal decision rule based on h^k achieves the smallest possible error probability, where the performance is averaged over the possible $\mathcal{C}_\mathcal{Y}$ generated from an RIE. In turn, we denote $\xi_i^X \leftrightarrow f_i$ as the corresponding information vector, and define the matrix $\boldsymbol{\Xi}^X \triangleq [\boldsymbol{\xi}_1^X \ \cdots \ \boldsymbol{\xi}_k^X]$.

Theorem 1 (Universal Feature Selection). *For $v, v' \in \mathcal{V}$, let $E_{h^k}(v, v')$ be the error exponent associated with the pairwise error probability distinguishing v and v' based on h^k, then the expected error exponent over a given RIE defined on the set of ϵ-configurations is given by*

$$\mathbb{E}\big[E_{h^k}(v, v')\big] = \frac{C_0}{2} \cdot \left\| \tilde{\mathbf{B}}\boldsymbol{\Xi}^X \big((\boldsymbol{\Xi}^X)^{\mathrm{T}}\boldsymbol{\Xi}^X\big)^{-\frac{1}{2}} \right\|_{\mathrm{F}}^2 + o(\epsilon^2), \tag{7}$$

where $C_0 \triangleq \frac{1}{4|\mathcal{Y}|} \cdot \mathbb{E}\Big[\big\|\boldsymbol{\phi}_v^{Y|V} - \boldsymbol{\phi}_{v'}^{Y|V}\big\|^2\Big]$ is independent of the choices of f_i's, and the expectations $\mathbb{E}[\cdot]$ are taken over this RIE.

Proof. See Appendix A. □

As a result of (7), designing the $\boldsymbol{\xi}_i^X$ as the singular vectors $\boldsymbol{\psi}_i^X$ of $\tilde{\mathbf{B}}$, for $i = 1, \ldots, k$, optimizes (7) for all RIEs, pairs of (v, v'), and ϵ-configurations. Thus, the feature functions corresponding to $\boldsymbol{\psi}_i^X$ are *universally optimal* for inferring the unknown attribute V. Moreover, (7) naturally leads to an information metric $\left\|\tilde{\mathbf{B}}\boldsymbol{\Xi}^X\left(\left(\boldsymbol{\Xi}^X\right)^{\mathrm{T}}\boldsymbol{\Xi}^X\right)^{-\frac{1}{2}}\right\|_{\mathrm{F}}^2$ for any feature $\boldsymbol{\Xi}^X$ of X, measured by projecting the normalized $\boldsymbol{\Xi}^X$ through a linear projection $\tilde{\mathbf{B}}$. This information metric quantifies how informative a feature of X is when solving inference problems with respect to Y and is optimized when designing features by singular vectors of $\tilde{\mathbf{B}}$. Thus, we can interpret the universal feature selection as solving the most informative features for data inferences via the SVD of $\tilde{\mathbf{B}}$, which also coincides with the maximally correlated features in (3). Later, we will show that the feature selection in DNNs shares the same information metric as universal feature selection in the local analysis regime.

3.2. Feature Extraction in Deep Neural Networks

3.2.1. Network with Ideal Expressive Power

For convenience of analysis, we first consider the ideal case where the neural network can express any feature mapping $s(\cdot)$ as desired. While this assumption can be rather strong, the existence of such ideal networks is guaranteed by the universal approximation theorem [40]. In addition, one goal of practical network designs is to approximate the ideal networks and obtain sufficient expressive power. For such networks, we will show that when X, Y are ϵ-dependent, the extracted feature $s(x)$ and weights $v(y)$ coincide with the solutions of the universal feature selection.

To begin, we use P_{XY} to denote the joint empirical distribution of the labeled samples $(x_i, y_i), i = 1, \ldots, N$, and P_X, P_Y to denote the corresponding marginal distributions. Then, the objective function of (5) is the empirical average of the log-likelihood function

$$\frac{1}{N}\sum_{i=1}^{N}\log \tilde{P}_{Y|X}(y_i|x_i) = \mathbb{E}_{P_{XY}}\left[\log \tilde{P}_{Y|X}(Y|X)\right].$$

Therefore, maximizing this empirical average is equivalent as minimizing the KL divergence:

$$(s^*, v^*, b^*) = \underset{(s,v,b)}{\arg\min}\ D(P_{XY}\|P_X\,\tilde{P}_{Y|X}^{(s,v,b)}). \tag{8}$$

This can be interpreted as finding the best fitting to empirical joint distribution P_{XY} by distributions of the form $P_X\,\tilde{P}_{Y|X}^{(s,v,b)}$. In our development, it is more convenient to denote the bias by $d(y) = b(y) - \log P_Y(y)$, for $y \in \mathcal{Y}$. Then, the following lemma illustrates the explicit constraint on the problem (8) in the local analysis regime.

Lemma 2. *If X, Y are ϵ-dependent, then the optimal v, d for (8) satisfy*

$$|\tilde{v}^{\mathrm{T}}(y)s(x) + \tilde{d}(y)| = O(\epsilon), \quad \textit{for all } x \in \mathcal{X},\ y \in \mathcal{Y}. \tag{9}$$

Proof. See Appendix B. □

In turn, we take (9) as the constraint for solving the problem (8) in the local analysis regime. Moreover, we define the information vectors for zero-mean vectors $\tilde{s}$, $\tilde{v}$ as $\xi^X(x) = \sqrt{P_X(x)}\,\tilde{s}(x)$, $\xi^Y(y) = \sqrt{P_Y(y)}\,\tilde{v}(y)$, and define matrices

$$\boldsymbol{\Xi}^Y \triangleq \left[\xi^Y(1) \quad \cdots \quad \xi^Y(|\mathcal{Y}|)\right]^{\mathrm{T}}, \quad \boldsymbol{\Xi}^X \triangleq \left[\xi^X(1) \quad \cdots \quad \xi^X(|\mathcal{X}|)\right]^{\mathrm{T}}.$$

Lemma 3. *The KL divergence (8) in the local analysis regime (9) can be expressed as*

$$D(P_{XY}\|P_X\,\tilde{P}_{Y|X}^{(s,v,b)}) = \frac{1}{2}\|\tilde{\mathbf{B}} - \boldsymbol{\Xi}^Y\left(\boldsymbol{\Xi}^X\right)^{\mathrm{T}}\|_{\mathrm{F}}^2 + \frac{1}{2}\eta^{(v,b)}(s) + o(\epsilon^2), \tag{10}$$

where $\eta^{(v,b)}(s) \triangleq \mathbb{E}_{P_Y}\left[(\mu_s^{\mathrm{T}}\tilde{v}(Y) + \tilde{d}(Y))^2\right]$.

Proof. See Appendix C. □

Lemma 3 reveals key insights for feature selection in neural networks. To see this, we consider the following two learning problems: learning the optimal weight v for given s and learning the optimal feature s for given v.

For the case that s is fixed, we can optimize (10) with Ξ^X fixed and obtain the following optimal weights:

Theorem 2. *For fixed* Ξ^X *and* μ_s*, the optimal* Ξ^{Y*} *to minimize* (10) *is given by*

$$\Xi^{Y*} = \tilde{\mathbf{B}}\,\Xi^X\left(\left(\Xi^X\right)^{\mathrm{T}}\Xi^X\right)^{-1}, \tag{11}$$

and the optimal weights $\tilde{v}^*$ *and bias* $\tilde{d}^*$ *are*

$$\tilde{v}^*(y) = \mathbb{E}_{P_{X|Y}}\left[\mathbf{\Lambda}^{-1}_{\tilde{s}(X)}\,\tilde{s}(X)\;\middle|\;Y = y\right], \quad \tilde{d}^*(y) = -\mu_s^{\mathrm{T}}\,\tilde{v}(Y). \tag{12}$$

where $\mathbf{\Lambda}_{\tilde{s}(X)}$ *denotes the covariance matrix of* $\tilde{s}(X)$.

Proof. See Appendix D. □

Specifically, when $s(x) = x$, Theorem 2 gives the optimal weights for softmax regression. Note that Equation (11) can be viewed as a projection of the input feature $\tilde{s}(x)$, to a feature $v(y)$ computable from the value of y, which is the most correlated feature to $\tilde{s}(x)$. The solution is given by the operation that left multiplies $\tilde{\mathbf{B}}$ matrix, which we refer to as *forward feature projection*.

Remark 1. *While we assume the continuous input* $s(x)$ *is a function of a discrete variable* X*, we only need the labeled samples between* s *and* Y *to compute the weights and bias from the conditional expectation* (12)*, and the correlation between* X *and* s *is irrelevant. Thus, our analysis for weights and bias can be applied to continuous input networks by just ignoring* X *and taking* s *as the real input to network.*

We then consider the "backward feature projection" problem, which attempts to find informative feature $s^*(X)$ to minimize the loss (10) with given weights and bias. In particular, we can show that the solution of this backward feature projection is precisely symmetric to the forward one.

Theorem 3. *For fixed* Ξ^Y *and* $\tilde{d}$*, the optimal* Ξ^{X*} *to minimize* (10) *is given by*

$$\Xi^{X*} = \tilde{\mathbf{B}}^{\mathrm{T}}\,\Xi^Y\left(\left(\Xi^Y\right)^{\mathrm{T}}\Xi^Y\right)^{-1}, \tag{13}$$

and the optimal feature function s^**, which are decomposed to* $\tilde{s}^*$ *and* μ_s^**, is given by*

$$\begin{aligned}\tilde{s}^*(x) &= \mathbb{E}_{P_{Y|X}}\left[\mathbf{\Lambda}^{-1}_{\tilde{v}(Y)}\,\tilde{v}(Y)\middle|X = x\right],\\ \mu_s^* &= -\mathbf{\Lambda}^{-1}_{\tilde{v}(Y)}\,\mathbb{E}_{P_Y}\left[\tilde{v}(Y)\,\tilde{d}(Y)\right],\end{aligned} \tag{14}$$

where $\mathbf{\Lambda}_{\tilde{v}(Y)}$ *denotes the covariance matrix of* $\tilde{v}(Y)$.

Proof. See Appendix D. □

Finally, when both s and (v, b) (and hence Ξ^X, Ξ^Y, d) can be designed, the optimal (Ξ^Y, Ξ^X) corresponds to the low rank factorization of $\tilde{\mathbf{B}}$, and the solutions coincide with the universal feature selection.

Theorem 4. *The optimal solutions for weights and bias to minimize* (10) *are given by* $\tilde{d}(y) = -\mu_s^{\mathrm{T}}\tilde{v}(y)$*, and* $(\boldsymbol{\Xi}^Y, \boldsymbol{\Xi}^X)^*$ *chosen as the largest k left and right singular vectors of* $\tilde{\mathbf{B}}$*.*

Proof. See Appendix E. □

Therefore, we conclude that the learning of neural networks, when both s and (v, b) are designable, is to extract the most correlated aspects of the input data X and the label Y that are informative features for data inferences from universal feature selection.

In the practical learning process of DNN, the BackProp updates the weights of the softmax layer and those on the previous layer(s) in an iterative manner. As we have illustrated in Lemma 3, such iterative updates will converge to the same solution as the alternating between the forward feature projection (11) and the backward feature projection (13), which is indeed the power method to solve the SVD for $\tilde{\mathbf{B}}$ [41], also known as the Alternating Conditional Expectation (ACE) algorithm [36].

Remark 2. *From Theorem 4, for a neural network with sufficient expressive power, the trained feature depends only on the distribution of input data rather than the training process. It is worth mentioning that this result does not contradict the practice that trained weights in hidden layers can be different during each training run. In fact, due to the over-parameterized nature of practical network designs, there exist multiple choices of weights in hidden layers to express the same optimal feature* $s(x)$*.*

3.2.2. Network with Restricted Expressive Power

The analysis of the previous section has considered neural networks with ideal expressive power, where the feature $s(X)$ can be selected as any desired function. In general, however, the form of feature functions that can be generalized is often limited by the network structure. In the following, we consider networks with restricted expressive power to characterize the impacts of network structure on the extracted feature.

For illustration, we consider the neural network with a hidden layer of k nodes, and a zero-mean continuous input $t = [t_1 \ \cdots \ t_m]^{\mathrm{T}} \in \mathbb{R}^m$ to this hidden layer, where t is assumed to be a function $t(x)$ of some discrete variable X. Our goal is to analyze the weights and bias in this layer with labeled samples $(t(x_i), y_i)$. Assume the activation function of the hidden layer is a generally smooth function $\sigma(\cdot)$, then the output $s_z(X)$ of the z-th hidden node is

$$s_z(x) = \sigma\Big(w^{\mathrm{T}}(z)t(x) + c(z)\Big), \quad \text{for } z = 1, \ldots, k, \ x \in \mathcal{X}, \tag{15}$$

where $w(z) \in \mathbb{R}^m$ and $c(z) \in \mathbb{R}$ are the weights and bias from input layer to hidden layer as shown in Figure 2. We denote $s = [s_1 \ \cdots \ s_k]^{\mathrm{T}}$ as the input vector to the output classification layer.

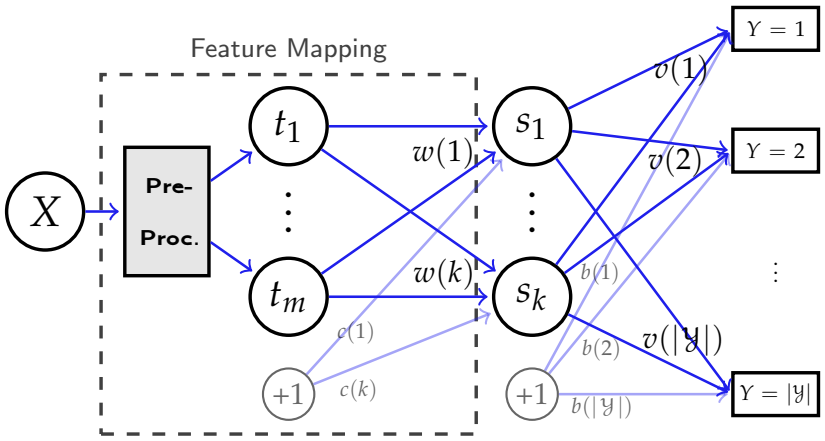

Figure 2. A multi-layer neural network, where the expressive power of the feature mapping $s(\cdot)$ is restricted by the hidden representation t. All hidden layers previous to t are fixed, represented by the "pre-processing" module.

To interpret the feature selection in hidden layers, we fix $(v(y), b(y))$ at the output layer and consider the problem of designing $(w(z), c(z))$ to minimize the loss function (8) at the output layer. Ideally, we should have picked $w(z)$ and $c(z)$ to generate $s(x)$ to match $s^*(x)$ from (14), which minimizes the loss. However, here we have the constraint that $s(x)$ must take the form of (15) and, intuitively, the network should select $w(z), c(z)$ so that $s(x)$ is close to $s^*(x)$. Our goal is to quantify the notion of such closeness.

To develop insights on feature selection in hidden layers, we again focus on the local analysis regime, where the weights and bias are assumed to satisfy the local constraint

$$\left|\tilde{v}^{\mathrm{T}}(y)s(x) + \tilde{d}(y)\right| = O(\epsilon), \quad \left|w^{\mathrm{T}}(z)\tilde{t}(x)\right| = O(\epsilon), \ \forall x, y, z. \tag{16}$$

Then, since t is zero-mean, we can express (15) as

$$s_z(x) = \sigma\Big(w^{\mathrm{T}}(z)t(x) + c(z)\Big) = w^{\mathrm{T}}(z)\tilde{t}(x) \cdot \sigma'(c(z)) + \sigma(c(z)) + o(\epsilon), \tag{17}$$

Moreover, we define a matrix $\tilde{\mathbf{B}}_1$ with the (z, x)th entry $\tilde{\mathbf{B}}_1(z, x) = \frac{\sqrt{P_X(x)}}{\sigma'(c(z))}\tilde{s}^*_z(x)$, which can be interpreted as a generalized CDM for the hidden layer. Furthermore, we denote $\xi^X_1(x) = \sqrt{P_X(x)}\,\tilde{t}(x)$ as the information vector of $\tilde{t}(x)$ with the matrix Ξ^X_1 defined as $\Xi^X_1 \triangleq \left[\xi^X_1(1) \quad \cdots \quad \xi^X_1(|\mathcal{X}|)\right]^{\mathrm{T}}$, and we also define

$$\mathbf{W} \triangleq \left[w(1) \quad \cdots \quad w(k)\right]^{\mathrm{T}}, \tag{18}$$

$$\mathbf{J} \triangleq \mathrm{diag}\{\sigma'(c(1)), \sigma'(c(2)), \cdots, \sigma'(c(k))\}. \tag{19}$$

The following theorem characterizes the loss (8).

Theorem 5. *Given the weights and bias (v, b) at the output layer, and for any input feature s, we denote $\mathcal{L}(s)$ as the loss (8) evaluated with respect to (v, b) and s. Then, with the constraints (16)*

$$\mathcal{L}(s) - \mathcal{L}(s^*) = \frac{1}{2}\left\|\Theta\tilde{\mathbf{B}}_1 - \Theta\mathbf{W}\left(\Xi^X_1\right)^{\mathrm{T}}\right\|^2_{\mathrm{F}} + \frac{1}{2}\kappa^{(v,b)}(s, s^*) + o(\epsilon^2), \tag{20}$$

where $\Theta \triangleq \left(\left(\Xi^Y\right)^{\mathrm{T}}\Xi^Y\right)^{1/2}\mathbf{J}$, and the term $\kappa^{(v,b)}(s, s^*) = (\mu_s - \mu_{s^*})^{\mathrm{T}}\mathbf{\Lambda}_{\tilde{v}(Y)}(\mu_s - \mu_{s^*})$.

Proof. See Appendix F. □

Equation (20) quantifies the closeness between s and s^* in terms of the loss (8). Then, our goal is to minimize (20), which can be separated to two optimization problems:

$$\mathbf{W}^* = \arg\min_{\mathbf{W}} \left\| \Theta \tilde{\mathbf{B}}_1 - \Theta \mathbf{W} \left(\Xi_1^X \right)^{\mathrm{T}} \right\|_{\mathrm{F}}^2, \tag{21}$$

$$\mu_s^* = \arg\min_{\mu_s} \kappa^{(v,b)}(s, s^*). \tag{22}$$

Note that the optimization problem (21) is similar to the one that appeared in Lemma 3, and the optimal solution is given by $\mathbf{W}^* = \tilde{\mathbf{B}}_1 \Xi_1^X \left(\left(\Xi_1^X \right)^{\mathrm{T}} \Xi_1^X \right)^{-1}$. Therefore, solving the optimal weights in the hidden layer can be interpreted as projecting $\tilde{s}^*(x)$ to the subspace of feature functions spanned by $t(x)$ to find the closest expressible function. In addition, the problem (22) is to choose μ_s (and hence the bias $c(z)$) to minimize the quadratic term similar to $\eta^{(v,b)}(s)$ in (10). Similar to the analyses of parameters in the last layer, we can obtain analytical solutions for hidden layer parameters, e.g., μ_s^* and w^*, with detailed discussions provided in Appendix G.

Overall, we observe the correspondence between (11), (14), and (21), (22), and interpret both operations as feature projections. Our argument can be generalized to any intermediate layer in a multi-layer network, with all the previous layers viewed as the fixed pre-processing that specifies $t(x)$, and all the layers after determining s^*. Then, the iterative procedure in back-propagation can be viewed as alternating projection finding the fixed-point solution over the entire network. This final fixed-point solution, even under the local assumption, might not be the SVD solution as in Theorem 4. This is because the limited expressive power of the network often makes it impossible to generate the desired feature function. In such cases, the concept of feature projection can be used to quantify this gap, and thus to measure the quality of the selected features.

3.3. Scoring Neural Networks

Given a learning problem, it is useful to tell whether or not some extracted features are informative [42]. Our previous development naturally gives rise to a performance metric.

Definition 5. *Given a feature $s(x) \in \mathbb{R}^k$ and weight $v(y) \in \mathbb{R}^k$ with the corresponding information matrices Ξ^X and Ξ^Y, the H-score $H(s, v)$ is defined as*

$$H(s,v) \triangleq \frac{1}{2} \|\tilde{\mathbf{B}}\|_{\mathrm{F}}^2 - \frac{1}{2} \|\tilde{\mathbf{B}} - \Xi^Y \left(\Xi^X \right)^{\mathrm{T}} \|_{\mathrm{F}}^2 = \mathbb{E}_{P_{XY}} \left[\tilde{s}^{\mathrm{T}}(X)\, \tilde{v}(Y) \right] - \frac{1}{2} \operatorname{tr} \left(\mathbf{\Lambda}_{\tilde{s}(X)} \mathbf{\Lambda}_{\tilde{v}(Y)} \right). \tag{23}$$

In addition, for given $s(x)$, we define the single-sided H-score $H(s)$ as

$$H(s) \triangleq \max_v H(s,v) \tag{24}$$

$$= \frac{1}{2} \|\tilde{\mathbf{B}}\|_{\mathrm{F}}^2 - \frac{1}{2} \|\tilde{\mathbf{B}} - \tilde{\mathbf{B}}\, \Xi^X \left(\left(\Xi^X \right)^{\mathrm{T}} \Xi^X \right)^{-1} \left(\Xi^X \right)^{\mathrm{T}} \|_{\mathrm{F}}^2 \tag{25}$$

$$= \frac{1}{2} \|\tilde{\mathbf{B}} \Xi^X \left(\left(\Xi^X \right)^{\mathrm{T}} \Xi^X \right)^{-\frac{1}{2}} \|_{\mathrm{F}}^2 = \frac{1}{2} \mathbb{E}_{P_Y} \left[\left\| \mathbb{E}_{P_{X|Y}} \left[\mathbf{\Lambda}_{\tilde{s}(X)}^{-1/2} \tilde{s}(X) \;\middle|\; Y \right] \right\|^2 \right]. \tag{26}$$

H-score can be used to measure the quality of features generated at any intermediate layer of the network. It is related to (20) when choosing the optimal bias and Θ as the identity matrix. This can be understood as taking the output of this layer $s(x)$ and directly feeding it to a softmax output layer with $v(y)$ used as the weights, and $H(s, v)$ measures the resulting performance. Note that $v(y)$ here can be an arbitrary function of Y, not necessarily the weights on the next layer computed by the network. When the optimal $v^*(y)$ as defined in (12) is used, the resulting performance becomes the one-sided H-score $H(s)$, which measures the quality of $s(x)$. In addition, by comparing (26) with (7), the performance measure $H(s)$ also coincides with the information metric (7), up to a scale factor.

Specifically, for a given dataset and a feature extractor that generate $s(\cdot)$, the H-score $H(s)$ can be efficiently computed from the second equation of (26). In addition, when we use H-score to compare the performance of different feature extractors (models), the model complexity has to be taken into account to reduce overfitting. To this end, we adopt Akaike information criterion (AIC) and define *AIC-corrected H-score*

$$H_{\mathrm{AIC}}(s) \triangleq H(s) - \frac{n_{\mathrm{p}}}{n_{\mathrm{s}}} \tag{27}$$

for comparing different models, where n_{p} and n_{s} represent the number of parameters in the model and the training sample size, respectively.

In current practice, the cross-entropy $\mathbb{E}_{P_{XY}}\left[\log \tilde{P}_{Y|X}^{(v,b)}\right]$ is often used as the performance metric. One can, in principle, also use log-loss to measure the effectiveness of the selected feature at the output of an intermediate layer [42]. However, one problem of this metric is that, for a given problem, it is not clear what value of log-loss one should expect, as the log-loss is generally unbounded. In contrast, the H-score can be directly computed from the data samples and has a clear upper bound. Indeed, it follows from Lemma 1 that, for k-dimensional feature s and weights v, we have the sequence of inequalities

$$H(s,v) \le H(s) \le \frac{1}{2}\sum_{i=1}^{k}\sigma_i^2 \le \frac{k}{2}, \tag{28}$$

where σ_i indicates the ith singular value of $\tilde{\mathbf{B}}$.

In particular, the first "$\le$" follows from the definition (24), and the gap between $H(s,v)$ and $H(v)$ measures the optimality of the weights v; the second "$\le$" follows from the first equality of (26), and the gap between two sides characterizes the difference between the chosen feature and the optimal solution, which is a useful measure of how restrictive (lack of expressive power) the network structure is; the last "$\le$" follows from the fact that $\sigma_i \le 1$ (cf. Lemma 1), which measures the dependency between data variable and label for the given dataset. In Section 3.4.3, we validate this metric on real data.

3.4. Experiments

This section presents experiments for validating our theoretical characterizations, with corresponding code available at https://github.com/XiangxiangXu/dnn (accessed on 7 December 2021). Specifically, all DNN models used in Section 3.4.3 are available at https://keras.io/applications/ (accessed on 7 December 2021).

3.4.1. Experimental Validation of Theorem 4

We first validate Theorem 4, the optimal feature extracted by network with ideal expressive power. Here, we consider the discrete data with alphabet sizes, $|\mathcal{X}| = 8$ and $|\mathcal{Y}| = 6$, and construct the network as shown in Figure 3. Specifically, the network input is the one-hot encoding of X, i.e., $[\mathbb{1}_X(1), \ldots, \mathbb{1}_X(|\mathcal{X}|)]^{\mathrm{T}}$, where $\mathbb{1}_X(x)$ takes one if and only if $X = x$, and takes zero otherwise. Then, the feature $s(X)$ is generated by a linear layer, with sigmoid function used as the activation function. For ease of comparison and presentation, we set feature dimension to $k = 1$, since otherwise the optimal feature (cf. Theorem 4) lies in a subspace and is non-unique. It can be verified that this network has ideal expressive power, i.e., with proper weights in the first layer, $s(X)$ can express any desired function up to scaling and shifting.

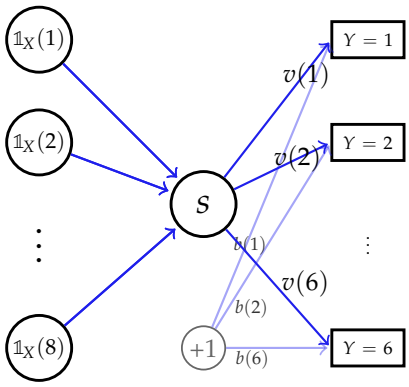

Figure 3. A simple neural network with ideal expressive power, which can generate any $k = 1$ dimensional feature s of X by tuning the weights in the first layer.

To compare the result trained by the neural network and that in Theorem 4, we first randomly generate a distribution P_{XY}, and then draw independently $n = 100{,}000$ pairs of (X, Y) samples. We then train the network using batch gradient descent, where we have applied Nesterov momentum [43] with the momentum hyperparameter being 0.9. In addition, we set the learning rate to 4 with a decay factor of 0.01 and clip gradients with norm exceeding 0.5. After training, the learned values of $s(x), v(y)$ and $b(y)$ are shown in Figure 4 and compared with theoretical results. From the figure, we can observe that the training results match our theoretical analyses.

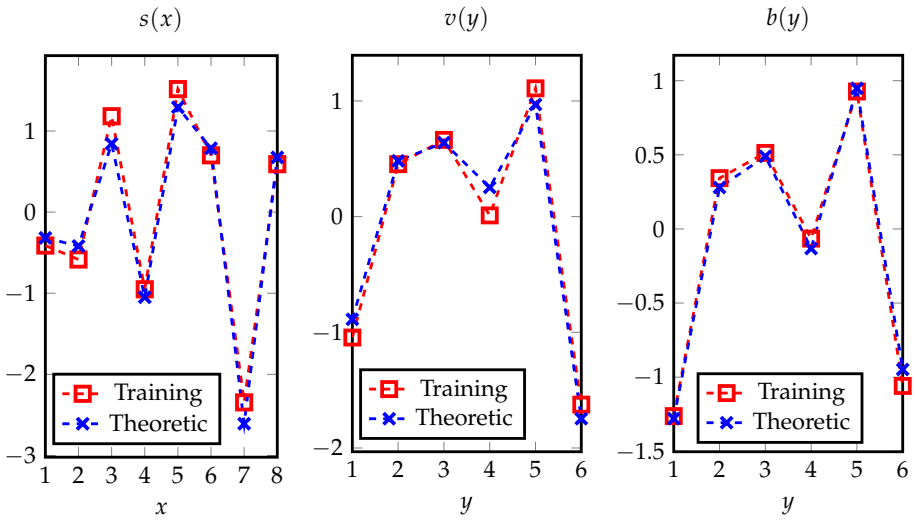

Figure 4. The trained feature s, weights v, and bias b of the network in Figure 3, which are compared with the corresponding theoretical results to show their coincidences.

3.4.2. Experimental Validation of Theorem 5

In addition, we validate Theorem 5 by the neural network depicted in Figure 5, with the same settings of X, Y. Specifically, the number of neurons in hidden layers are set to $m = 4$ and $k = 3$, where $t(X)$ is randomly generated from X, and we have chosen sigmoid function as the activation function $\sigma(\cdot)$ to generate $s(x)$. We then fix the weights and bias at the output layer and train the weights $w(1), w(2), w(3)$ and bias c in the hidden layer to optimize the log-loss. Specifically, we use the batch gradient descent with the Nesterov momentum hyperparameter being 0.9. In addition, we set the learning rate to 4 with a decay factor of 10^{-6} and clip gradients with norm exceeding 0.1. After training, Figure 6 shows the matching between the learned results and the corresponding theoretical values.

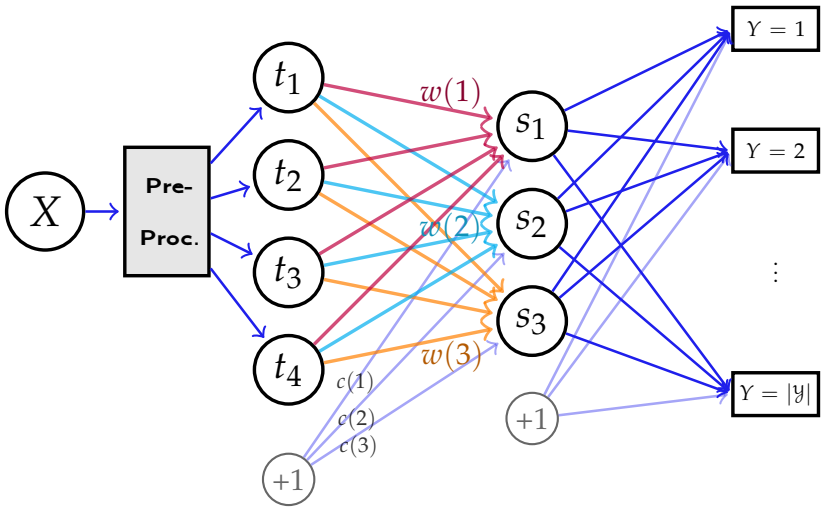

Figure 5. The designed network for validating the impact of network structure on feature extraction, with $m = 4$ and $k = 3$ neurons in two hidden layers. Our goal is to compare the learned weights $w(1), w(2), w(3)$ and bias c in the hidden layer with our theoretic characterizations in Section 3.2.2.

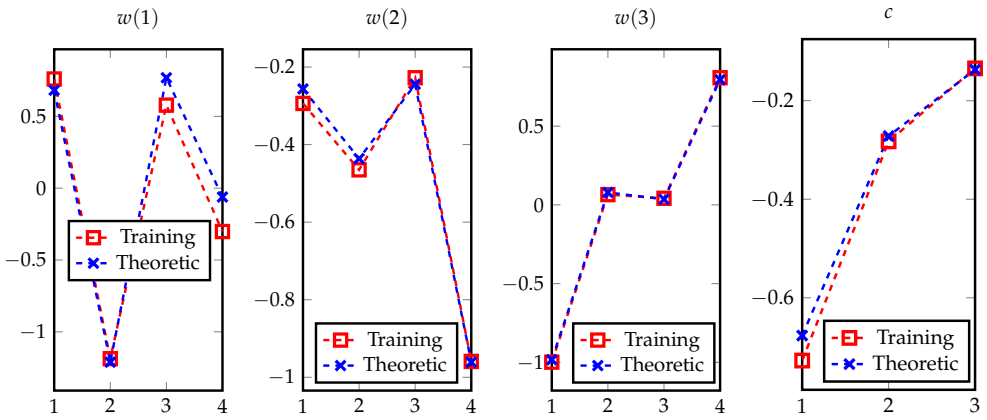

Figure 6. The trained weights w and bias c of the network in Figure 5, which are compared with the corresponding theoretical results to show their coincidences.

3.4.3. Experimental Validation of H-Score

To validate H-score as a performance measure for extracted features, we compare the H-score and classification accuracy of DNNs on image classification tasks. Specifically, we use the ImageNet Large Scale Visual Recognition Challenge 2012 (ILSVRC2012) [22] dataset as the dataset and extract features using several deep neural networks with representative architectures designs [44–49]. After training the feature extractors on the ILSVRC2012 training set, we then compute the H-score of the feature in the last hidden layer, as well as the classification accuracies on ILSVRC2012 validation set (here, we use ILSVRC2012 validation set for testing, as the labels in ILSVRC2012 testing set have not been publicly released). The results are summarized in Table 1, where $H_{\mathrm{AIC}}(s)$ is the AIC-corrected H-score as defined in (27), with n_{p} being the number of model parameters, and $n_{\mathrm{s}} = 1{,}300{,}000$ corresponding to the number of training samples in ImageNet. The AIC-corrected H-score is consistent with the classification accuracy, which validates the effectiveness of H-score as a measurement of neural networks.

Table 1. Classification accuracy and H-score for different DNN models on ImageNet dataset, where "Paras" indicates the number of parameters (in millions) in the model and H_{AIC} represents the AIC-corrected H-score.

DNN Model	Paras [$\times 10^6$]	$H(s)$	$H_{\text{AIC}}(s)$	Accuracy [%]
VGG16 [44]	138.4	148.3	41.9	64.2
VGG19 [44]	143.7	152.7	42.2	64.7
MobileNet [45]	4.3	45.9	42.6	68.4
DenseNet121 [46]	8.1	59.5	53.3	71.4
DenseNet169 [46]	14.3	81.2	70.2	73.6
DenseNet201 [46]	20.2	89.1	73.5	74.4
Xception [47]	22.9	179.8	162.2	77.5
InceptionV3 [48]	23.9	181.2	162.9	76.3
InceptionResNetV2 [49]	55.9	241.1	198.1	79.1

4. Discussion

Our characterization gives an information-theoretic interpretation of the feature extraction process in DNNs, which also provides a practical performance measure for scoring neural networks. Different from empirical studies focusing on specific datasets [7], our development is based on the probability distribution space, which is more general and can also provide theoretic insights. Moreover, the information-theoretic framework allows us to obtain direct operational meaning and better interpretations for the solutions, compared with optimization-based theoretical characterizations, e.g., [11,13].

As a first step in establishing a rigorous framework for DNN analysis, the present work can be extended in both theoretical and practical aspects. From the theoretical perspective, one extension is to investigate the analytical properties for general DNNs, using the theoretic insights obtained from local analysis regime. For example, it was shown in [50] that the symmetry between feature and weights in DNNs established in the local analysis regime (cf. Section 3.2.1) also holds for general probability distributions. Another extension is to apply the framework to investigate the optimal feature for structured data or network, e.g., data with sparsity structure [51].

From the practical perspective, in addition to the demonstrated example of evaluating existing DNN models (cf. Section 3.4.3), the H-score can also be used as an objective function in designing learning algorithms. In particular, such usages have been illustrated in multi-modal learning [52] and transfer learning [53] tasks.

5. Conclusions

In this paper, we apply the local information geometric analysis and provide an information-theoretic interpretation to the feature extraction scheme in DNNs. We first establish an information metric for features in inference tasks by formalizing the information-theoretic feature selection problem. In addition, we demonstrate that the features extracted by DNNs coincide with the information-theoretically optimal feature, with the same metric measuring the performance of features, called H-score. Furthermore, we discuss the usage of the H-score for measuring the effectiveness of DNNs. Our framework demonstrates a connection between the practical deep learning implementations and information-theoretic characterizations, which can provide theoretical insights for DNN analysis and learning algorithm designs.

Author Contributions: X.X., S.-L.H., L.Z. and G.W.W. contributed to the conceptualization, methodology, and writing of this paper. All authors have read and agreed to the published version of the manuscript.

Funding: The work of S.-L. Huang was supported in part by the National Natural Science Foundation of China under Grant 61807021 and the Shenzhen Science and Technology Program under Grant KQTD20170810150821146. The work of L. Zheng was supported in part by the National Science

Foundation (NSF) under Award CNS-2002908 and the Office of Naval Research (ONR) under Grant N00014-19-1-2621.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

Appendix A. Proof of Theorem 1

We commence with the characterization of the error exponent.

Lemma A1. *Given a reference distribution $P_X \in \text{relint}(\mathcal{P}^{\mathcal{X}})$, a constant $\epsilon > 0$ and integers n and k, let $x_1, \ldots, x_n$ denote i.i.d. samples from one of P_1 or P_2, where $P_1, P_2 \in \mathcal{N}_\epsilon^{\mathcal{X}}(P_X)$. To decide whether P_1 or P_2 is the generating distribution, a sequence of k-dimensional statistics $h^k = (h_1, \ldots, h_k)$ is constructed as*

$$h_i = \frac{1}{n}\sum_{l=1}^{n} f_i(x_l), \quad i = 1, \ldots, k, \tag{A1}$$

where $(f_1(X), \ldots, f_k(X))$ are zero mean, unit-variance, and uncorrelated with respect to P_X, i.e.,

$$\mathbb{E}_{P_X}[f_i(X)] = 0, \quad i \in \{1, \ldots, k\} \tag{A2}$$

$$\mathbb{E}_{P_X}\big[f_i(X) f_j(X)\big] = \delta_{ij}, \quad i, j \in \{1, \ldots, k\}. \tag{A3}$$

Then, the error probability of the decision based on h^k decays exponentially in n as $n \to \infty$, with (Chernoff) exponent

$$\lim_{n\to\infty} \frac{-\log p_e}{n} \triangleq E_{h^k} = \sum_{i=1}^{k} E_{h_i}, \tag{A4}$$

where

$$E_{h_i} = \frac{1}{8}\langle \phi_1 - \phi_2, \xi_i \rangle^2 + o(\epsilon^2), \tag{A5}$$

and $\phi_1 \leftrightarrow P_1, \phi_2 \leftrightarrow P_2, \xi_i \leftrightarrow f_i(X), i \in \{1, \ldots, k\}$ are the corresponding information vectors.

Proof of Lemma A1. Since the rule is to decide based on comparing the projection

$$\sum_{i=1}^{k} h_i \big(\mathbb{E}_{P_1}[f_i(X)] - \mathbb{E}_{P_2}[f_i(X)]\big)$$

to a threshold, via Cramér's theorem [54], the error exponent under P_j $(j = 1, 2)$ is

$$E_j(\lambda) = \min_{P \in \mathcal{S}(\lambda)} D(P \| P_j), \tag{A6}$$

where

$$\mathcal{S}(\lambda) \triangleq \Big\{ P \in \mathcal{P}^{\mathcal{X}} : \mathbb{E}_P\big[f^k(X)\big] = \lambda\, \mathbb{E}_{P_1}\big[f^k(X)\big] + (1-\lambda)\, \mathbb{E}_{P_2}\big[f^k(X)\big] \Big\}. \tag{A7}$$

Now, since (A2) holds, we obtain

$$\begin{aligned}
\mathbb{E}_{P_j}[f_i(X)] &= \sum_{x\in\mathcal{X}} P_j(x)\, f_i(x) \\
&= \sum_{x\in\mathcal{X}} P_X(x)\, f_i(x) + \sum_{x\in\mathcal{X}} (P_j(x) - P_X(x)) f_i(x) \\
&= \mathbb{E}_{P_X}[f_i(X)] + \sum_{x\in\mathcal{X}} \sqrt{P_X(x)}\, \phi_j(x) \cdot \frac{\xi_i(x)}{\sqrt{P_X(x)}} \\
&= \sum_{x\in\mathcal{X}} \phi_j(x)\, \xi_i(x)
\end{aligned}$$

$$= \langle \phi_j, \xi_i \rangle, \quad j = 1,2 \text{ and } i = 1, \ldots, k, \tag{A8}$$

which we express compactly as

$$\mathbb{E}_{P_j}\left[f^k(X)\right] = \langle \phi_j, \xi^k \rangle, \quad j = 1,2$$

with $\xi^k \triangleq (\xi_1, \ldots, \xi_k)$.

Hence, the constraint (A7) is expressed in information vectors as

$$\langle \phi, \xi_i \rangle = \langle \lambda\, \phi_1 + (1-\lambda)\, \phi_2, \xi_i \rangle, \quad i = 1, \cdots, k,$$

i.e.,

$$\langle \phi, \xi^k \rangle = \langle \lambda\, \phi_1 + (1-\lambda)\, \phi_2, \xi^k \rangle. \tag{A9}$$

In turn, the optimal P in (A6), which we denoted by P^*, lies in the exponential family through P_j with natural statistic $f^k(x)$, i.e., the k-dimensional family whose members are of the form

$$\log \tilde{P}_{\theta^k}(x) = \sum_{i=1}^{k} \theta_i f_i(x) + \log P_j(x) - \alpha\left(\theta^k\right),$$

for which the associated information vector is

$$\tilde{\phi}_{\theta^k}(x) = \sum_{i=1}^{k} \theta_i \xi_i(x) + \phi_j(x) - \alpha(\theta^k)\sqrt{P_X(x)} + o(\epsilon), \tag{A10}$$

where we have used the fact that

$$\begin{aligned}
\log Q_X(x) &= \log P_X(x) + \log \frac{Q_X(x)}{P_X(x)} \\
&= \log P_X(x) + \log\left(1 + \frac{1}{\sqrt{P_X(x)}}\phi(x)\right) \\
&= \log P_X(x) + \frac{1}{\sqrt{P_X(x)}}\phi(x) + o(\epsilon)
\end{aligned}$$

for all $Q_X \in \mathcal{N}_\epsilon^{\mathcal{X}}(P_X)$ with the information vector $\phi \leftrightarrow Q_X$. As a result,

$$\langle \tilde{\phi}_{\theta^k}, \xi_i \rangle = \theta_i + \langle \phi_j, \xi_i \rangle + o(\epsilon),$$

where we have used (A3). Hence, via (A9), we obtain that the intersection with the linear family (A7) is at $P^* = P_{\theta^{k*}}$ with

$$\theta_i^* = \langle \lambda\phi_1 + (1-\lambda)\phi_2 - \phi_j, \xi_i \rangle + o(\epsilon)$$

and thus

$$\begin{aligned}
E_j(\lambda) &= D(P^* \| P_j) \\
&= \frac{1}{2}\left\|\tilde{\phi}_{\theta^k} - \phi_j\right\|^2 + o(\epsilon^2) && \text{(A11)} \\
&= \frac{1}{2}\left\|\sum_{i=1}^{k} \theta_i^* \xi_i\right\|^2 + \frac{1}{2}\alpha\left(\theta^{k*}\right)^2 + o(\epsilon^2) && \text{(A12)} \\
&= \frac{1}{2}\sum_{i=1}^{k} (\theta_i^*)^2 + \frac{1}{2}\alpha\left(\theta^{k*}\right)^2 + o(\epsilon^2) && \text{(A13)} \\
&= \frac{1}{2}\sum_{i=1}^{k} \langle \lambda\phi_1 + (1-\lambda)\phi_2 - \phi_j, \xi_i \rangle^2 + o(\epsilon^2), && \text{(A14)}
\end{aligned}$$

where to obtain (A11) we have exploited the local approximation of KL divergence [18], to obtain (A12) we have exploited (A10), to obtain (A13) we have again exploited (A3), and to obtain (A14) we have used that

$$\alpha\left(\theta^{k^*}\right) = o(\epsilon^2)$$

since $\theta^{k^*} = O(\epsilon)$ and

$$\alpha(0) = 0, \quad \text{and} \quad \nabla\alpha(0) = \mathbb{E}_{P_j}\left[f^k(X)\right] = \langle \phi_j, \xi^k \rangle = O(\epsilon).$$

Finally, $E_1(\lambda) = E_2(\lambda)$ when $\lambda = 1/2$, so the overall error probability has exponent (A5). □

Then, the following lemma demonstrates a property of information vectors in a Markov chain.

Lemma A2. *Given the Markov relation $X \leftrightarrow Y \leftrightarrow V$ and any $v \in \mathcal{V}$, let $\boldsymbol{\phi}_v^{X|V}$ and $\boldsymbol{\phi}_v^{Y|V}$ denote the associated information vectors for $P_{X|V}(\cdot|v)$ and $P_{Y|V}(\cdot|v)$, then we have*

$$\boldsymbol{\phi}_v^{X|V} = \tilde{\mathbf{B}}^{\mathrm{T}} \boldsymbol{\phi}_v^{Y|V}. \tag{A15}$$

Proof of Lemma A2. From the Markov relation we have

$$P_X(x) = \sum_{y \in \mathcal{Y}} P_{X|Y}(x|y) P_Y(y)$$

and

$$P_{X|V}(x|v) = \sum_{y \in \mathcal{Y}} P_{X|Y,V}(x|y,v) P_{Y|V}(y|v) = \sum_{y \in \mathcal{Y}} P_{X|Y}(x|y) P_{Y|V}(y|v).$$

As a result,

$$P_{X|V}(x|v) - P_X(x) = \sum_{y \in \mathcal{Y}} P_{X|Y}(x|y)[P_{Y|V}(y|v) - P_Y(y)],$$

from which we obtain the corresponding information vector

$$\begin{aligned}
\phi_v^{X|V}(x) &= \frac{1}{\sqrt{P_X(x)}} \sum_{y \in \mathcal{Y}} P_{X|Y}(x|y) \sqrt{P_Y(y)} \phi_v^{Y|V}(y) \\
&= \sum_{y \in \mathcal{Y}} \left[\tilde{\mathbf{B}}(y,x) + \sqrt{P_X(x) P_Y(y)} \right] \phi_v^{Y|V}(y) \\
&= \sum_{y \in \mathcal{Y}} \tilde{\mathbf{B}}(y,x) \phi_v^{Y|V}(y),
\end{aligned} \tag{A16}$$

where the last equality follows from the fact that

$$\sum_{y \in \mathcal{Y}} \sqrt{P_Y(y)} \phi_v^{Y|V}(y) = \sum_{y \in \mathcal{Y}} [P_{Y|V}(y|v) - P_Y(y)] = 0.$$

Finally, rewrite (A16) in the matrix form and we obtain (A15). □

In addition, the following lemma is useful for dealing with the expectation over an RIE.

Lemma A3. *Let* $\mathbf{z}$ *be a spherically symmetric random vector of dimension* M*, i.e., for any orthogonal* $\mathbf{Q}$ *we have* $\mathbf{z} \overset{\mathrm{d}}{=} \mathbf{Q}\mathbf{z}$*. If* $\mathbf{A}$ *is a fixed matrix of compatible dimensions, then*

$$\mathbb{E}\left[\|\mathbf{z}^{\mathrm{T}}\mathbf{A}\|^2\right] = \frac{1}{M}\mathbb{E}\left[\|\mathbf{z}\|^2\right]\|\mathbf{A}\|_{\mathrm{F}}^2. \tag{A17}$$

Proof of Lemma A3. By definition we have $\mathbf{\Lambda_z} = \mathbf{Q}\mathbf{\Lambda_z}\mathbf{Q}^{\mathrm{T}}$ for any orthogonal $\mathbf{Q}$; hence, $\mathbf{\Lambda_z}$ is diagonal. Suppose $\mathbf{\Lambda_z} = \lambda\,\mathbf{I}$, then from

$$\mathrm{tr}(\mathbf{\Lambda_z}) = \mathbb{E}\left[\|\mathbf{z}\|^2\right] = \lambda M$$

we obtain

$$\lambda = \frac{1}{M}\,\mathrm{tr}(\mathbf{\Lambda_z}).$$

As a result, we have

$$\mathbb{E}\left[\|\mathbf{z}^{\mathrm{T}}\mathbf{A}\|^2\right] = \mathrm{tr}\left(\mathbf{A}^{\mathrm{T}}\mathbf{\Lambda_z}\mathbf{A}\right) = \lambda\,\mathrm{tr}\left(\mathbf{A}^{\mathrm{T}}\mathbf{A}\right) = \frac{1}{M}\mathbb{E}\left[\|\mathbf{z}\|^2\right]\|\mathbf{A}\|_{\mathrm{F}}^2.$$

□

Proceeding to our proof of Theorem 1, by definition of feature functions, we have $\mathbb{E}_{P_X}\left[f_i(X)\right] = 0, i = 1, \ldots, k$. Suppose $\boldsymbol{f}$ is the vector representation of f^k and denote by $\tilde{\boldsymbol{f}} \triangleq \mathbf{\Lambda}_f^{-1/2}\boldsymbol{f}$ the normalized $\boldsymbol{f}$, with $\mathbf{\Lambda}_f^{1/2}$ denoting any square root matrix of $\mathbf{\Lambda}_f$. Then, the corresponding statistics $\tilde{f}^k = (\tilde{f}_1, \ldots, \tilde{f}_k)$ satisfy the constraints (A2) and (A3). In addition, we construct the statistic $\tilde{h}^k = (\tilde{h}_1, \ldots, \tilde{h}_k)$ as [cf. (A1)]

$$\tilde{h}_i = \frac{1}{n}\sum_{l=1}^{n}\tilde{f}_i(x_l), \quad i = 1, \ldots, k. \tag{A18}$$

Then, from Lemma A1, the error exponent of distinguishing v and v' based on $\tilde{h}^k$ is

$$\begin{aligned} E_{\tilde{h}^k}(v, v') &= \frac{1}{8}\sum_{i=1}^{k}\left[\left(\boldsymbol{\phi}_v^{X|V} - \boldsymbol{\phi}_{v'}^{X|V}\right)^{\mathrm{T}}\tilde{\boldsymbol{\xi}}_i^X\right]^2 + o(\epsilon^2) \\ &= \frac{1}{8}\left\|\left(\boldsymbol{\phi}_v^{X|V} - \boldsymbol{\phi}_{v'}^{X|V}\right)^{\mathrm{T}}\tilde{\mathbf{\Xi}}^X\right\|^2 + o(\epsilon^2), \end{aligned}$$

where $\boldsymbol{\phi}_v^{X|V}$ denotes the associated information vector for $P_{X|V}(\cdot|v)$, $\tilde{\boldsymbol{\xi}}_i^X$ denotes the information vectors of $\tilde{f}_i$, and $\tilde{\mathbf{\Xi}}^X \triangleq [\tilde{\boldsymbol{\xi}}_1^X, \ldots, \tilde{\boldsymbol{\xi}}_k^X]$. Since the optimal decision rule is linear, the error exponent is invariant with linear transformations of statistics, i.e.,

$$\begin{aligned} E_{h^k}(v, v') = E_{\tilde{h}^k}(v, v') &= \frac{1}{8}\left\|\left(\boldsymbol{\phi}_v^{X|V} - \boldsymbol{\phi}_{v'}^{X|V}\right)^{\mathrm{T}}\tilde{\mathbf{\Xi}}^X\right\|^2 + o(\epsilon^2) \\ &= \frac{1}{8}\left\|\left(\boldsymbol{\phi}_v^{Y|V} - \boldsymbol{\phi}_{v'}^{Y|V}\right)^{\mathrm{T}}\tilde{\mathbf{B}}\tilde{\mathbf{\Xi}}^X\right\|^2 + o(\epsilon^2), \end{aligned} \tag{A19}$$

where the last equality follows from Lemma A2.

As a result, taking the expectation of (A19) over a given RIE yields

$$\begin{aligned} \mathbb{E}\left[E_{h^k}(v, v')\right] &= \frac{1}{8}\mathbb{E}\left[\left\|\left(\boldsymbol{\phi}_v^{Y|V} - \boldsymbol{\phi}_{v'}^{Y|V}\right)^{\mathrm{T}}\tilde{\mathbf{B}}\tilde{\mathbf{\Xi}}^X\right\|^2\right] + o(\epsilon^2) \\ &= \frac{\mathbb{E}\left[\left\|\boldsymbol{\phi}_v^{Y|V} - \boldsymbol{\phi}_{v'}^{Y|V}\right\|^2\right]}{8|\mathcal{Y}|}\left\|\tilde{\mathbf{B}}\tilde{\mathbf{\Xi}}^X\right\|_{\mathrm{F}}^2 + o(\epsilon^2), \end{aligned}$$

where we have exploited Lemma A3. Finally, the error exponent (7) can be obtained via noting from the definition of $\tilde{f}^k$ that

$$\tilde{\Xi}^X = \Xi^X \left((\Xi^X)^{\mathrm{T}} \Xi^X \right)^{-\frac{1}{2}}.$$

Appendix B. Proof of Lemma 2

We first prove two useful lemmas.

Lemma A4. *For distributions $P \in \mathrm{relint}(\mathcal{P}^{\mathcal{X}})$, $Q, R \in \mathcal{P}^{\mathcal{X}}$, and sufficiently small ϵ, if $D(P\|Q) \leq \epsilon^2$ and $D(P\|R) \leq \epsilon^2$, then there exists a constant $C > 0$ independent of ϵ, such that $D(Q\|R) \leq C\epsilon^2$.*

Proof of Lemma A4. Denote by $\|\cdot\|_1$ the ℓ_1-distance between distributions, i.e., $\|P - Q\|_1 \triangleq \sum_{x\in\mathcal{X}} |P(x) - Q(x)|$, then from Pinsker's inequality [14], we have

$$\|P - Q\|_1 \leq \sqrt{2D(P\|Q)} < \sqrt{2}\epsilon, \tag{A20}$$

$$\|P - R\|_1 \leq \sqrt{2D(P\|R)} < \sqrt{2}\epsilon, \tag{A21}$$

which implies

$$\|Q - R\|_1 \leq \|P - Q\|_1 + \|P - R\|_1 \leq 2\sqrt{2}\epsilon. \tag{A22}$$

In addition, with $p_{\min} \triangleq \min_{x\in\mathcal{X}} P(x)$, for all $x \in \mathcal{X}$ we have

$$R(x) > P(x) - |P(x) - R(x)| \tag{A23}$$

$$> \min_{x\in\mathcal{X}} P(x) - \sqrt{2}\epsilon \tag{A24}$$

$$= p_{\min} - \sqrt{2}\epsilon, \tag{A25}$$

where to obtain (A24) we have used (A21). Note that since $P \in \mathrm{relint}(\mathcal{P}^{\mathcal{X}})$ we have $p_{\min} > 0$, and thus $R(x) > p_{\min}/2$ for sufficiently small ϵ. As a result,

$$D(Q\|R) \leq \sum_{x\in\mathcal{X}} \frac{(Q(x) - R(x))^2}{R(x)} \tag{A26}$$

$$\leq \frac{2}{p_{\min}} \sum_{x\in\mathcal{X}} [Q(x) - R(x)]^2 \tag{A27}$$

$$\leq \frac{2\|Q - R\|_1^2}{p_{\min}} \tag{A28}$$

$$\leq \frac{16}{p_{\min}} \epsilon^2, \tag{A29}$$

where to obtain (A26) we have used the fact that KL divergence is upper bounded by corresponding χ^2-divergence [55], and to obtain (A29) we have used (A22). □

Lemma A5. *For all $(x, y) \in \mathcal{X} \times \mathcal{Y}$, we have*

$$D(P_X P_Y \| P_X \tilde{P}_{Y|X}^{(s,v,b)}) \geq P_X(x) \log \left[P_Y(y) e^{\tau(x,y)} + (1 - P_Y(y)) e^{-\frac{P_Y(y)}{1-P_Y(y)}\tau(x,y)} \right]$$

where $\tilde{P}_{Y|X}^{(s,v,b)}$ is as defined in (4), and where we have defined $\tau(x, y) \triangleq \tilde{v}^{\mathrm{T}}(y)s(x) + \tilde{d}(y)$.

Proof of Lemma A5. First, we can rewrite the conditional distribution $\tilde{P}_{Y|X}^{(s,v,b)}(y|x)$ as

$$\begin{aligned}\tilde{P}_{Y|X}^{(s,v,b)}(y|x) = \frac{e^{v^{\mathrm{T}}(y)s(x)+b(y)}}{\sum_{y'\in\mathcal{Y}} e^{v^{\mathrm{T}}(y')s(x)+b(y')}} &= \frac{P_Y(y)e^{v^{\mathrm{T}}(y)s(x)+d(y)}}{\sum_{y'\in\mathcal{Y}} P_Y(y')e^{v^{\mathrm{T}}(y')s(x)+d(y')}} \\ &= \frac{P_Y(y)e^{\tilde{v}^{\mathrm{T}}(y)s(x)+\tilde{d}(y)}}{\sum_{y'\in\mathcal{Y}} P_Y(y')e^{\tilde{v}^{\mathrm{T}}(y')s(x)+\tilde{d}(y')}} \\ &= \frac{P_Y(y)e^{\tau(x,y)}}{\sum_{y'\in\mathcal{Y}} P_Y(y')e^{\tau(x,y')}}. \end{aligned} \tag{A30}$$

Then, the KL divergence $D(P_XP_Y\|P_X\,\tilde{P}_{Y|X}^{(s,v,b)})$ can be expressed as

$$\begin{aligned} D(P_XP_Y\|P_X\,\tilde{P}_{Y|X}^{(s,v,b)}) &= \sum_{(x,y)\in\mathcal{X}\times\mathcal{Y}} P_X(x)P_Y(y)\log\frac{\sum_{y'\in\mathcal{Y}} P_Y(y')e^{\tau(x,y')}}{e^{\tau(x,y)}} \\ &= \sum_{x\in\mathcal{X}} P_X(x)\log\left[\sum_{y'\in\mathcal{Y}} P_Y(y')e^{\tau(x,y')}\right] - \mathbb{E}_{P_XP_Y}[\tau(X,Y)] \\ &= \sum_{x\in\mathcal{X}} P_X(x)\log\left[\sum_{y'\in\mathcal{Y}} P_Y(y')e^{\tau(x,y')}\right], \end{aligned} \tag{A31}$$

where to obtain the last equality we have used the fact $\mathbb{E}_{P_XP_Y}[\tau(X,Y)] = 0$. As a result, we have

$$D(P_XP_Y\|P_X\,\tilde{P}_{Y|X}^{(s,v,b)}) \geq P_X(x)\log\left[\sum_{y'\in\mathcal{Y}} P_Y(y')e^{\tau(x,y')}\right] \tag{A32}$$

$$\geq P_X(x)\log\left[P_Y(y)e^{\tau(x,y)} + (1-P_Y(y))e^{-\frac{P_Y(y)}{1-P_Y(y)}\tau(x,y)}\right], \tag{A33}$$

where the last inequality follows from Jensen's inequality:

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

□

Proceeding to our proof of Lemma 2, first note that when $v = d = 0$, we have $\tilde{P}_{Y|X}^{(s,v,b)} = P_Y$. As a result, the optimal v, d for (8) satisfy

$$\begin{aligned} D(P_{XY}\|P_X\,\tilde{P}_{Y|X}^{(s,v,b)}) &\leq D(P_{XY}\|P_XP_Y) \\ &\leq \sum_{(x,y)\in\mathcal{X}\times\mathcal{Y}} \frac{[P_{X,Y}(x,y) - P_X(x)P_Y(y)]^2}{P_X(x)P_Y(y)} \\ &\leq \epsilon^2, \end{aligned} \tag{A34}$$

where to obtain the second inequality we have again exploited χ^2-divergence as an upper bound of KL divergence [55], and to obtain the last inequality we have used the definition of ϵ-dependency.

As $P_{XY} \in \text{relint}(\mathcal{P}^{\mathcal{X}\times\mathcal{Y}})$, from Lemma A4, there exist $C > 0$ and $\epsilon_1 > 0$ such that $D(P_X P_Y \| P_X \tilde{P}_{Y|X}^{(s,v,b)}) < C\epsilon^2$ for all $\epsilon < \epsilon_1$. Furthermore, from Lemma A5, for all $(x,y) \in \mathcal{X}\times\mathcal{Y}$ and $\epsilon \in (0,\epsilon_1)$, we have

$$C\epsilon^2 \geq P_X(x) \log\left[P_Y(y)e^{\tau(x,y)} + (1-P_Y(y))e^{-\frac{P_Y(y)}{1-P_Y(y)}\tau(x,y)}\right]. \tag{A35}$$

Note that the right-hand side of (A35) satisfies

$$\log\left[P_Y(y)e^{\tau(x,y)} + (1-P_Y(y))e^{-\frac{P_Y(y)}{1-P_Y(y)}\tau(x,y)}\right] = \frac{P_Y(y)}{2(1-P_Y(y))}\tau^2(x,y) + o(\tau^2(x,y)).$$

Therefore, there exists $\delta > 0$ independent of ϵ_1, such that for all $|\tau(x,y)| \leq \delta$, we have

$$\log\left[P_Y(y)e^{\tau(x,y)} + (1-P_Y(y))e^{-\frac{P_Y(y)}{1-P_Y(y)}\tau(x,y)}\right] > \frac{P_Y(y)}{2}\tau^2(x,y). \tag{A36}$$

In addition, if $|\tau(x,y)| > \delta$, we have

$$\begin{aligned}
&\log\left[P_Y(y)e^{\tau(x,y)} + (1-P_Y(y))e^{-\frac{P_Y(y)}{1-P_Y(y)}\tau(x,y)}\right] \\
&\geq \min\left\{\log\left[P_Y(y)e^{\delta} + (1-P_Y(y))e^{-\frac{P_Y(y)}{1-P_Y(y)}\delta}\right], \log\left[P_Y(y)e^{-\delta} + (1-P_Y(y))e^{\frac{P_Y(y)}{1-P_Y(y)}\delta}\right]\right\} \\
&\geq \frac{P_Y(y)}{2}\delta^2,
\end{aligned}$$

where to obtain the second inequality we have exploited the monotonicity of function $t \mapsto P_Y(y)e^t + (1-P_Y(y))e^{-\frac{P_Y(y)}{1-P_Y(y)}t}$, and to obtain the third inequality we have exploited (A36).

As a result, we have

$$\log\left[P_Y(y)e^{\tau(x,y)} + (1-P_Y(y))e^{-\frac{P_Y(y)}{1-P_Y(y)}\tau(x,y)}\right] > \frac{P_Y(y)}{2}\cdot\min\{\delta^2, \tau^2(x,y)\}. \tag{A37}$$

Hence, (A35) becomes

$$C\epsilon^2 \geq \frac{P_X(x)P_Y(y)}{2}\cdot\min\{\delta^2, \tau^2(x,y)\}, \tag{A38}$$

from which we can obtain $\tau(x,y) = O(\epsilon)$. To see this, let

$$\epsilon_2 \triangleq \frac{\delta}{\sqrt{2C}}\cdot\min_{(x,y)\in\mathcal{X}\times\mathcal{Y}}\sqrt{P_X(x)P_Y(y)}, \quad \epsilon_0 \triangleq \min\{\epsilon_1, \epsilon_2\}.$$

Then, for all $\epsilon < \epsilon_0$, we have

$$C\epsilon^2 < \frac{P_X(x)P_Y(y)}{2}\cdot\delta^2,$$

and (A38) implies $|\tau(x,y)| < C'\epsilon$ with $C' = \sqrt{\frac{2C}{P_X(x)P_Y(y)}}$.

Appendix C. Proof of Lemma 3

Proof. From Lemma 2, there exists $C' > 0$ such that for all $(x,y) \in \mathcal{X}\times\mathcal{Y}$, we have

$$|\tilde{v}^{\mathrm{T}}(y)s(x) + \tilde{d}(y)| < C'\epsilon, \tag{A39}$$

which implies

$$|\mu_s^{\mathrm{T}}\tilde{v}(y)+\tilde{d}(y)|<C\epsilon, \tag{A40}$$

$$|\tilde{v}^{\mathrm{T}}(y)\tilde{s}(x)|<2C\epsilon, \tag{A41}$$

with $C=\max\{C',1\}$.

From (A30), we can assume $\mathbb{E}_{P_Y}[v(Y)]=\mathbb{E}_{P_Y}[d(Y)]=0$ without loss of generality. Then, (4) can be rewritten as

$$\tilde{P}_{Y|X}^{(s,v,b)}(y|x)=\frac{P_Y(y)e^{\tilde{v}^{\mathrm{T}}(y)s(x)+\tilde{d}(y)}}{\sum_{y'\in\mathcal{Y}}P_Y(y')e^{\tilde{v}^{\mathrm{T}}(y')s(x)+\tilde{d}(y')}}, \tag{A42}$$

and the numerator can be written as

$$\begin{aligned}P_Y(y)e^{\tilde{v}^{\mathrm{T}}(y)s(x)+\tilde{d}(y)}&=P_Y(y)\Big(1+\tilde{v}^{\mathrm{T}}(y)s(x)+\tilde{d}(y)+o(\epsilon)\Big)\\&=P_Y(y)\Big(1+\tilde{v}^{\mathrm{T}}(y)s(x)+\tilde{d}(y)\Big)+o(\epsilon),\end{aligned}$$

where we have used (A39). Similarly, from

$$\begin{aligned}\sum_{y'\in\mathcal{Y}}P_Y(y)e^{\tilde{v}^{\mathrm{T}}(y)s(x)+\tilde{d}(y)}&=\sum_{y'\in\mathcal{Y}}P_Y(y)\Big(1+\tilde{v}^{\mathrm{T}}(y)s(x)+\tilde{d}(y)\Big)+o(\epsilon)\\&=1+\mathbb{E}_{P_Y}\Big[\tilde{v}^{\mathrm{T}}(Y)s(x)\Big]+\mathbb{E}_{P_Y}\big[\tilde{d}(y)\big]+o(\epsilon)\\&=1+o(\epsilon)\end{aligned}$$

we obtain

$$\frac{1}{\sum_{y'\in\mathcal{Y}}P_Y(y)e^{\tilde{v}^{\mathrm{T}}(y)s(x)+\tilde{d}(y)}}=\frac{1}{1+o(\epsilon)}=1+o(\epsilon).$$

As a result, (A42) can be written as

$$\begin{aligned}\tilde{P}_{Y|X}^{(s,v,b)}(y|x)&=\Big[P_Y(y)\Big(1+\tilde{v}^{\mathrm{T}}(y)s(x)+\tilde{d}(y)\Big)+o(\epsilon)\Big][1+o(\epsilon)]\\&=P_Y(y)\Big(1+\tilde{v}^{\mathrm{T}}(y)s(x)+\tilde{d}(y)\Big)+o(\epsilon),\end{aligned} \tag{A43}$$

which implies $P_X\,\tilde{P}_{Y|X}^{(v,b)}\in\mathcal{N}_{C\epsilon}^{\mathcal{X}\times\mathcal{Y}}(P_XP_Y)$ for sufficiently small ϵ. In addition, the local assumption of distributions implies that $P_{XY}\in\mathcal{N}_{\epsilon}^{\mathcal{X}\times\mathcal{Y}}(P_XP_Y)\subset\mathcal{N}_{C\epsilon}^{\mathcal{X}\times\mathcal{Y}}(P_XP_Y)$. Again, from the local approximation of KL divergence [18]

$$D(P_1\|P_2)=\frac{1}{2}\|\phi_1-\phi_2\|^2+o(\epsilon^2), \tag{A44}$$

we have

$$\begin{aligned}&D(P_{Y,X}\|P_X\,\tilde{P}_{Y|X}^{(s,v,b)})\\&=\frac{1}{2}\sum_{x\in\mathcal{X},y\in\mathcal{Y}}\frac{\Big[P_{Y,X}(y,x)-\tilde{P}_{Y|X}^{(s,v,b)}(y|x)P_X(x)\Big]^2}{P_Y(y)P_X(x)}+o(\epsilon^2)\\&=\frac{1}{2}\sum_{x\in\mathcal{X},y\in\mathcal{Y}}\Bigg[\frac{P_{Y,X}(y,x)}{\sqrt{P_Y(y)P_X(x)}}-\sqrt{P_Y(y)P_X(x)}\\&\quad-\sqrt{P_Y(y)P_X(x)}\Big(\tilde{v}^{\mathrm{T}}(y)s(x)+\tilde{d}(y)+o(\epsilon)\Big)\Bigg]^2+o(\epsilon^2)\end{aligned}$$

$$
\begin{aligned}
&= \frac{1}{2}\sum_{x\in\mathcal{X},y\in\mathcal{Y}}\Big[\tilde{\mathbf{B}}(y,x) - \sqrt{P_Y(y)P_X(x)}\tilde{v}^{\mathrm{T}}(y)\tilde{s}(x) \\
&\quad -\sqrt{P_Y(y)P_X(x)}\Big(\tilde{d}(y)+\mu_s^{\mathrm{T}}\tilde{v}(y)\Big) - \sqrt{P_Y(y)P_X(x)}o(\epsilon)\Big]^2 + o(\epsilon^2) \\
&\overset{(*)}{=} \frac{1}{2}\sum_{x\in\mathcal{X},y\in\mathcal{Y}}\Big[\tilde{\mathbf{B}}(y,x) - \sqrt{P_Y(y)P_X(x)}\tilde{v}^{\mathrm{T}}(y)\tilde{s}(x)\Big]^2 \\
&\quad + \frac{1}{2}\sum_{x\in\mathcal{X},y\in\mathcal{Y}}\Big[\sqrt{P_Y(y)P_X(x)}\Big(\tilde{d}(y)+\mu_s^{\mathrm{T}}\tilde{v}(y)\Big)\Big]^2 + o(\epsilon^2) \\
&= \frac{1}{2}\sum_{x\in\mathcal{X},y\in\mathcal{Y}}\Big[\tilde{\mathbf{B}}(y,x) - \big(\xi^Y(y)\big)^{\mathrm{T}}\xi^X(x)\Big]^2 + \frac{1}{2}\mathbb{E}_{P_Y}\Big[\big(\tilde{d}(y)+\mu_s^{\mathrm{T}}\tilde{v}(y)\big)^2\Big] + o(\epsilon^2) \\
&= \frac{1}{2}\|\tilde{\mathbf{B}} - \Xi^Y\big(\Xi^X\big)^{\mathrm{T}}\|_{\mathrm{F}}^2 + \frac{1}{2}\eta^{(v,b)}(s) + o(\epsilon^2),
\end{aligned}
$$

where to obtain $(*)$, we have used (A40) and (A41) together with the fact $|\tilde{\mathbf{B}}(y,x)| < \epsilon$, and that

$$
\begin{aligned}
\sum_{x\in\mathcal{X},y\in\mathcal{Y}}\tilde{\mathbf{B}}(y,x)\sqrt{P_Y(y)P_X(x)}\Big(\tilde{d}(y)+\mu_s^{\mathrm{T}}\tilde{v}(y)\Big) &= 0, \\
\sum_{x\in\mathcal{X},y\in\mathcal{Y}}P_Y(y)P_X(x)\tilde{v}^{\mathrm{T}}(y)\tilde{s}(x)\Big(\tilde{d}(y)+\mu_s^{\mathrm{T}}\tilde{v}(y)\Big) &= 0,
\end{aligned}
$$

since $\mathbb{E}\big[\tilde{d}(Y)\big] = 0, \mathbb{E}[\tilde{s}(X)] = \mathbb{E}[\tilde{v}(Y)] = 0$. □

Appendix D. Proofs of Theorems 2 and 3

Theorems 2 and 3 can be proved based on Lemma 3.

Proofs of Theorems 2 and 3. Note that the value of $d(\cdot)$ only affects the second term of the KL divergence; hence, we can always choose $d(\cdot)$ such that $\tilde{d}(y)+\mu_s^{\mathrm{T}}\tilde{v}(y) = 0$. Then, the (Ξ^Y, Ξ^X) pair should be chosen as

$$
(\Xi^Y, \Xi^X)^* = \arg\min_{(\Xi^Y,\Xi^X)}\|\tilde{\mathbf{B}} - \Xi^Y\big(\Xi^X\big)^{\mathrm{T}}\|_{\mathrm{F}}^2. \tag{A45}
$$

Set the derivative (we use the denominator-layout notation of matrix calculus where the scalar-by-matrix derivative will have the same dimension as the matrix)

$$
\frac{\partial}{\partial\Xi^Y}\|\tilde{\mathbf{B}} - \Xi^Y\big(\Xi^X\big)^{\mathrm{T}}\|_{\mathrm{F}}^2 = 2(\Xi^Y\big(\Xi^X\big)^{\mathrm{T}}\Xi^X - \tilde{\mathbf{B}}\Xi^X) \tag{A46}
$$

to zero, and the optimal Ξ^Y for fixed Ξ^X is (here, we assume the matrix $\big(\Xi^X\big)^{\mathrm{T}}\Xi^X = \mathbf{\Lambda}_{\tilde{s}(X)}$ is invertible; for the case where $\big(\Xi^X\big)^{\mathrm{T}}\Xi^X$ is singular, we can obtain a similar result with ordinary matrix inverse replaced by the Moore–Penrose inverse)

$$
\Xi^{Y*} = \tilde{\mathbf{B}}\Xi^X\big(\big(\Xi^X\big)^{\mathrm{T}}\Xi^X\big)^{-1}. \tag{A47}
$$

As $\mathbf{1}^{\mathrm{T}}\sqrt{\mathbf{P}_Y}\,\tilde{\mathbf{B}} = 0$, we have $\mathbf{1}^{\mathrm{T}}\sqrt{\mathbf{P}_Y}\,\Xi^{Y*} = 0$, which demonstrates that Ξ^{Y*} is a valid matrix for a zero-mean feature vector.

To express Ξ^{Y*} of (A47) in the form of s and v, we can make use of the correspondence between feature and information vectors. We can show that, for a zero-mean feature function $f(X)$ with corresponding information vector ϕ, we have the correspondence

$\mathbb{E}_{P_{X|Y}}[f(X)|Y] \leftrightarrow \tilde{\mathbf{B}}\phi$. To see this, note that the y-th element of information vector $\tilde{\mathbf{B}}\phi$ is given by

$$\begin{aligned}\sum_{x\in\mathcal{X}} \tilde{\mathbf{B}}(y,x)\phi(x) &= \sum_{x\in\mathcal{X}} \frac{P_{XY}(x,y)-P_X(x)P_Y(y)}{\sqrt{P_X(x)P_Y(y)}} f(x)\sqrt{P_X(x)} \\ &= \frac{1}{\sqrt{P_Y(y)}} \sum_{x\in\mathcal{X}} P_{XY}(x,y) f(x) \\ &= \frac{1}{\sqrt{P_Y(y)}} \mathbb{E}_{P_{X|Y}}[f(X)|Y=y].\end{aligned}$$

Using similar methods, we can verify that $\mathbf{\Lambda}_{\tilde{s}(X)} = \left(\Xi^X\right)^{\mathrm{T}}\Xi^X$. As a result, (A47) is equivalent to

$$\tilde{v}^*(y) = \mathbb{E}_{P_{X|Y}}\left[\mathbf{\Lambda}^{-1}_{\tilde{s}(X)}\tilde{s}(X) \;\middle|\; Y=y\right]. \tag{A48}$$

By a symmetry argument, we can also obtain the first two equations of Theorem 3. To obtain the third equations of these two theorems, we need to minimize $\eta^{(v,b)}(s) = \mathbb{E}_{P_Y}\left[(\mu_s^{\mathrm{T}}\tilde{v}(Y)+\tilde{d}(Y))^2\right]$. For given $\tilde{v}$ and μ_s, the optimal $\tilde{d}$ is

$$\tilde{d}^*(y) = -\mu_s^{\mathrm{T}}\tilde{v}(Y), \tag{A49}$$

and the corresponding $\eta^{(v,b)}(s) = 0$.

In addition, for given $\tilde{d}$ and $\tilde{v}$, we have

$$\begin{aligned}\eta^{(v,b)}(s) &= \mathbb{E}_{P_Y}\left[(\mu_s^{\mathrm{T}}\tilde{v}(Y)+\tilde{d}(Y))^2\right] \\ &= \mu_s^{\mathrm{T}}\mathbf{\Lambda}_{\tilde{v}(Y)}\,\mu_s + 2\mu_s^{\mathrm{T}}\mathbb{E}_{P_Y}\left[\tilde{v}(Y)\tilde{d}(Y)\right] + \mathrm{var}(\tilde{d}(Y)).\end{aligned} \tag{A50}$$

Set $\frac{\partial}{\partial\mu_s}\eta^{(v,b)}(s) = 0$ and we obtain

$$\mu_s^* = -\mathbf{\Lambda}^{-1}_{\tilde{v}(Y)}\mathbb{E}_{P_Y}\left[\tilde{v}(Y)\tilde{d}(Y)\right]. \tag{A51}$$

□

Appendix E. Proof of Theorem 4

Proof. From Lemma 3, choosing the optimal (Ξ^Y, Ξ^X) is equivalent to solving the matrix factorization problem of $\tilde{\mathbf{B}}$. Since both Ξ^Y and Ξ^X have rank no greater than k, from the Eckart–Young–Mirsky theorem [56], the optimal choice of $\Xi^Y\left(\Xi^X\right)^{\mathrm{T}}$ should be the truncated singular value decomposition of $\tilde{\mathbf{B}}$ with top k singular values. As a result, $(\Xi^Y, \Xi^X)^*$ are the left and right singular vectors of $\tilde{\mathbf{B}}$ corresponding to the largest k singular values.

The optimality of bias $\tilde{d}(y) = -\mu_s^{\mathrm{T}}\tilde{v}(y)$ has already been shown in Appendix D. □

Appendix F. Proof of Theorem 5

The following lemma is useful to prove Theorem 5.

Lemma A6 (Pythagorean theorem). *Let Ξ^{X*} be the optimal matrix for given Ξ^Y as defined in (13). Then,*

$$\left\|\tilde{\mathbf{B}} - \Xi^Y\left(\Xi^X\right)^{\mathrm{T}}\right\|_{\mathrm{F}}^2 - \left\|\tilde{\mathbf{B}} - \Xi^Y\left(\Xi^{X*}\right)^{\mathrm{T}}\right\|_{\mathrm{F}}^2 = \left\|\Xi^Y\left(\Xi^{X*}\right)^{\mathrm{T}} - \Xi^Y\left(\Xi^X\right)^{\mathrm{T}}\right\|_{\mathrm{F}}^2. \tag{A52}$$

Proof of Lemma A6. Denote by $\langle \mathbf{U}, \mathbf{V} \rangle$ the Frobenius inner product of matrices $\mathbf{U}$ and $\mathbf{V}$, i.e., $\langle \mathbf{U}, \mathbf{V} \rangle \triangleq \mathrm{tr}(\mathbf{U}^{\mathrm{T}}\mathbf{V})$, and we have

$$\begin{aligned}\left\langle \tilde{\mathbf{B}} - \Xi^Y \left(\Xi^{X*}\right)^{\mathrm{T}}, \Xi^Y \left(\Xi^X\right)^{\mathrm{T}} \right\rangle &= \mathrm{tr}\left(\tilde{\mathbf{B}}\Xi^X \left(\Xi^Y\right)^{\mathrm{T}}\right) - \mathrm{tr}\left(\Xi^{X*}\left(\Xi^Y\right)^{\mathrm{T}}\Xi^Y\left(\Xi^X\right)^{\mathrm{T}}\right) \\ &= \mathrm{tr}\left(\tilde{\mathbf{B}}\Xi^X \left(\Xi^Y\right)^{\mathrm{T}}\right) - \mathrm{tr}\left(\tilde{\mathbf{B}}^{\mathrm{T}}\Xi^Y\left(\Xi^X\right)^{\mathrm{T}}\right) \\ &= 0.\end{aligned}$$

As a result, we obtain

$$\begin{aligned}\left\|\tilde{\mathbf{B}} - \Xi^Y \left(\Xi^X\right)^{\mathrm{T}}\right\|_{\mathrm{F}}^2 &= \left\|\tilde{\mathbf{B}} - \Xi^Y \left(\Xi^{X*}\right)^{\mathrm{T}} + \left(\Xi^Y \left(\Xi^{X*}\right)^{\mathrm{T}} - \Xi^Y \left(\Xi^X\right)^{\mathrm{T}}\right)\right\|_{\mathrm{F}}^2 \\ &= \left\|\tilde{\mathbf{B}} - \Xi^Y \left(\Xi^{X*}\right)^{\mathrm{T}}\right\|_{\mathrm{F}} + \left\|\Xi^Y \left(\Xi^{X*}\right)^{\mathrm{T}} - \Xi^Y \left(\Xi^X\right)^{\mathrm{T}}\right\|_{\mathrm{F}}^2 \\ &\quad + 2\left\langle \tilde{\mathbf{B}} - \Xi^Y \left(\Xi^{X*}\right)^{\mathrm{T}}, \Xi^Y\left(\left(\Xi^{X*}\right)^{\mathrm{T}} - \left(\Xi^X\right)^{\mathrm{T}}\right)\right\rangle \\ &= \left\|\tilde{\mathbf{B}} - \Xi^Y \left(\Xi^{X*}\right)^{\mathrm{T}}\right\|_{\mathrm{F}} + \left\|\Xi^Y \left(\Xi^{X*}\right)^{\mathrm{T}} - \Xi^Y \left(\Xi^X\right)^{\mathrm{T}}\right\|_{\mathrm{F}}^2,\end{aligned}$$

which finishes the proof. □

Proceeding to our proof of Theorem 5, from Lemma A6 we have

$$\begin{aligned}&\mathcal{L}(s) - \mathcal{L}(s^*) \\ &= \frac{1}{2}\left[\left\|\tilde{\mathbf{B}} - \Xi^Y \left(\Xi^X\right)^{\mathrm{T}}\right\|_{\mathrm{F}}^2 - \left\|\tilde{\mathbf{B}} - \Xi^Y \left(\Xi^{X*}\right)^{\mathrm{T}}\right\|_{\mathrm{F}}^2\right] + \frac{1}{2}\left[\eta^{(v,b)}(s) - \eta^{(v,b)}(s^*)\right] + o(\epsilon^2) \\ &= \frac{1}{2}\left\|\Xi^Y \left(\Xi^{X*}\right)^{\mathrm{T}} - \Xi^Y \left(\Xi^X\right)^{\mathrm{T}}\right\|_{\mathrm{F}}^2 + \frac{1}{2}\kappa^{(v,b)}(s, s^*) + o(\epsilon^2),\end{aligned}$$

where $\kappa^{(v,b)}(s, s^*) \triangleq \eta^{(v,b)}(s) - \eta^{(v,b)}(s^*)$. We then optimize $\left\|\Xi^Y \left(\Xi^{X*}\right)^{\mathrm{T}} - \Xi^Y \left(\Xi^X\right)^{\mathrm{T}}\right\|_{\mathrm{F}}^2$ and $\kappa^{(v,b)}(s, s^*)$ separately.

For the first term, we need to express Ξ^X in terms of $\mathbf{W}$ and Ξ_1^X. From (17), we obtain

$$\mathbb{E}[s_z(X)] = \sigma(c(z)) + o(\epsilon), \tag{A53}$$

$$\tilde{s}_z(x) = w^{\mathrm{T}}(z)\tilde{t}(x) \cdot \sigma'(c(z)) + o(\epsilon), \tag{A54}$$

which can be expressed in information vectors as

$$\Xi^X = \Xi_1^X \mathbf{W}^{\mathrm{T}}\mathbf{J} + o(\epsilon). \tag{A55}$$

From Theorem 3, we have

$$\Xi^{X*} = \tilde{\mathbf{B}}^{\mathrm{T}}\Xi^Y \left(\left(\Xi^Y\right)^{\mathrm{T}}\Xi^Y\right)^{-1}. \tag{A56}$$

As a result, we have

$$\begin{aligned}\left\|\Xi^Y \left(\Xi^{X*}\right)^{\mathrm{T}} - \Xi^Y \left(\Xi^X\right)^{\mathrm{T}}\right\|_{\mathrm{F}}^2 &= \left\|\left(\left(\Xi^Y\right)^{\mathrm{T}}\Xi^Y\right)^{1/2}\left(\left(\Xi^{X*}\right)^{\mathrm{T}} - \left(\Xi^X\right)^{\mathrm{T}}\right)\right\|_{\mathrm{F}}^2 \\ &= \left\|\left(\left(\Xi^Y\right)^{\mathrm{T}}\Xi^Y\right)^{1/2} \cdot \left(\left(\Xi^{X*}\right)^{\mathrm{T}} - \mathbf{J}\mathbf{W}\left(\Xi_1^X\right)^{\mathrm{T}} - o(\epsilon)\right)\right\|_{\mathrm{F}}^2 \\ &= \left\|\left(\left(\Xi^Y\right)^{\mathrm{T}}\Xi^Y\right)^{1/2} \cdot \left(\left(\Xi^{X*}\right)^{\mathrm{T}} - \mathbf{J}\mathbf{W}\left(\Xi_1^X\right)^{\mathrm{T}}\right)\right\|_{\mathrm{F}}^2 + o(\epsilon^2) \\ &= \left\|\left(\left(\Xi^Y\right)^{\mathrm{T}}\Xi^Y\right)^{1/2}\mathbf{J} \cdot \left(\mathbf{J}^{-1}\left(\Xi^{X*}\right)^{\mathrm{T}} - \mathbf{W}\left(\Xi_1^X\right)^{\mathrm{T}}\right)\right\|_{\mathrm{F}}^2 + o(\epsilon^2) \\ &= \left\|\Theta\tilde{\mathbf{B}}_1 - \Theta\mathbf{W}\left(\Xi_1^X\right)^{\mathrm{T}}\right\|_{\mathrm{F}}^2 + o(\epsilon^2),\end{aligned} \tag{A57}$$

where the third equality follows from the fact that [cf. (A41)] $\tilde{s}(x) = O(\epsilon)$ and $\tilde{v}(y) = O(1)$, and the last equality follows from the definitions $\tilde{\mathbf{B}}_1 \triangleq \mathbf{J}^{-1}\left(\Xi^{X*}\right)^{\mathrm{T}}$ and $\Theta \triangleq \left(\left(\Xi^Y\right)^{\mathrm{T}}\Xi^Y\right)^{1/2}\mathbf{J}$.

For the second term, from (A50) and (A51), we have

$$\begin{aligned}\kappa^{(v,b)}(s,s^*) &= [(\mu_s-\mu_{s^*})+\mu_{s^*}]^{\mathrm{T}}\mathbf{\Lambda}_{\tilde{v}(Y)}[(\mu_s-\mu_{s^*})+\mu_{s^*}]\\ &\quad -\mu_{s^*}^{\mathrm{T}}\mathbf{\Lambda}_{\tilde{v}(Y)}\mu_{s^*}+2(\mu_s-\mu_{s^*})^{\mathrm{T}}\mathbb{E}_{P_Y}[\tilde{v}(Y)\tilde{d}(Y)]\\ &= (\mu_s-\mu_{s^*})^{\mathrm{T}}\mathbf{\Lambda}_{\tilde{v}(Y)}(\mu_s-\mu_{s^*})+2(\mu_s-\mu_{s^*})^{\mathrm{T}}\left(\mathbf{\Lambda}_{\tilde{v}(Y)}\mu_{s^*}+\mathbb{E}_{P_Y}[\tilde{v}(Y)\tilde{d}(Y)]\right)\\ &= (\mu_s-\mu_{s^*})^{\mathrm{T}}\mathbf{\Lambda}_{\tilde{v}(Y)}(\mu_s-\mu_{s^*}).\end{aligned} \tag{A58}$$

Combining (A57) and (A58) finishes the proof.

Appendix G. Analyses of Hidden Layer Parameters

First, from (A53), the bias $c(z)$ of hidden layer is (when $\mu_t \neq 0$, the formula should be modified as $c(z)=\sigma^{-1}(\mu_s^*(z))-\mu_t^{\mathrm{T}}w+o(\epsilon)$.)

$$c(z)=\sigma^{-1}(\mu_s^*(z))+o(\epsilon).$$

To obtain μ_s^*, let us define $\sigma_{\min}\triangleq\inf_x\sigma(x), \sigma_{\max}\triangleq\sup_x\sigma(x)$. Then, the optimal μ_s is the solution of

$$\begin{aligned}\underset{\mu_s}{\text{minimize}}\quad & (\mu_s-\mu_{s^*})^{\mathrm{T}}\mathbf{\Lambda}_{\tilde{v}(Y)}(\mu_s-\mu_{s^*})\\ \text{subject to}\quad & \sigma_{\min}\preceq\mu_s\preceq\sigma_{\max}.\end{aligned} \tag{A59}$$

If μ_{s^*} satisfies the constraint of (A59), then it is the optimal solution. Otherwise, some elements of μ_s^* will become either $\sigma_{\min}$ or $\sigma_{\max}$, known as the saturation phenomenon [21].

To obtain $\mathbf{W}^*$, let

$$\begin{aligned}\tilde{\mathbf{B}}_1' &\triangleq \Theta\tilde{\mathbf{B}}_1=\left(\left(\Xi^Y\right)^{\mathrm{T}}\Xi^Y\right)^{-1/2}\left(\Xi^Y\right)^{\mathrm{T}}\tilde{\mathbf{B}},\\ \mathbf{W}' &\triangleq \Theta\mathbf{W}=\left(\left(\Xi^Y\right)^{\mathrm{T}}\Xi^Y\right)^{1/2}\mathbf{J}\mathbf{W}.\end{aligned}$$

Then, the optimal $\mathbf{W}'$ is given by

$$\mathbf{W}'^*=\arg\min_{\mathbf{W}'}\|\tilde{\mathbf{B}}_1'-\mathbf{W}'\left(\Xi_1^X\right)^{\mathrm{T}}\|_{\mathrm{F}}^2=\tilde{\mathbf{B}}_1'\Xi_1^X(\left(\Xi_1^X\right)^{\mathrm{T}}\Xi_1^X)^{-1}. \tag{A60}$$

Hence, $\mathbf{W}^*$ is given by

$$\begin{aligned}\mathbf{W}^*=\Theta^{-1}\mathbf{W}'^* &= \Theta^{-1}\tilde{\mathbf{B}}_1'\Xi_1^X(\left(\Xi_1^X\right)^{\mathrm{T}}\Xi_1^X)^{-1}\\ &= \tilde{\mathbf{B}}_1\Xi_1^X(\left(\Xi_1^X\right)^{\mathrm{T}}\Xi_1^X)^{-1}\\ &= \mathbf{J}^{-1}\cdot[\Xi^Y(\left(\Xi^Y\right)^{\mathrm{T}}\Xi^Y)^{-1}]^{\mathrm{T}}\tilde{\mathbf{B}}\,\Xi_1^X(\left(\Xi_1^X\right)^{\mathrm{T}}\Xi_1^X)^{-1},\end{aligned}$$

where the term $\tilde{\mathbf{B}}\,\Xi_1^X(\left(\Xi_1^X\right)^{\mathrm{T}}\Xi_1^X)^{-1}$ corresponds to a feature projection of $\tilde{t}(X)$:

$$\tilde{\mathbf{B}}\,\Xi_1^X\left(\left(\Xi_1^X\right)^{\mathrm{T}}\Xi_1^X\right)^{-1}\leftrightarrow\mathbb{E}_{P_{X|Y}}\left[\mathbf{\Lambda}_{\tilde{t}(X)}^{-1}\tilde{t}(X)\;\middle|\;Y\right]. \tag{A61}$$

As a consequence, this multi-layer neural network conducts a generalized feature projection between features extracted from different layers. Note that the projected feature $\mathbb{E}_{P_{\tilde{t}|Y}}\left[\mathbf{\Lambda}_{\tilde{t}}^{-1}\tilde{t}\middle|Y\right]$ depends only on the distribution $P_{\tilde{t}|Y}$ and does not depend on the distribution $P_{X|Y}$. Therefore, the above computations can be accomplished without knowing the hidden random variable X and can be applied to general cases.

References

1. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 770–778.
2. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Minneapolis, MN, USA, 3–5 June 2019; Volume 1 (Long and Short Papers); Association for Computational Linguistics: Minneapolis, MN, USA, 2019; pp. 4171–4186.
3. Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language Models are Few-Shot Learners. In *Advances in Neural Information Processing Systems*; Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M.F., Lin, H., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2020; Volume 33, pp. 1877–1901.
4. Silver, D.; Huang, A.; Maddison, C.J.; Guez, A.; Sifre, L.; Van Den Driessche, G.; Schrittwieser, J.; Antonoglou, I.; Panneershelvam, V.; Lanctot, M.; et al. Mastering the game of Go with deep neural networks and tree search. *Nature* **2016**, *529*, 484–489. [CrossRef]
5. Arulkumaran, K.; Cully, A.; Togelius, J. Alphastar: An evolutionary computation perspective. In Proceedings of the Genetic and Evolutionary Computation Conference Companion, Prague, Czech Republic, 13–17 July 2019; pp. 314–315.
6. MacKay, D.J.C. *Information Theory, Inference, and Learning Algorithms*; Cambridge University Press: Cambridge, UK, 2003; ISBN 9780521642989.
7. Zintgraf, L.M.; Cohen, T.S.; Adel, T.; Welling, M. Visualizing Deep Neural Network Decisions: Prediction Difference Analysis. In Proceedings of the 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, 24–26 April 2017.
8. Papyan, V.; Han, X.; Donoho, D.L. Prevalence of neural collapse during the terminal phase of deep learning training. *Proc. Natl. Acad. Sci. USA* **2020**, *117*, 24652–24663. [CrossRef]
9. Adebayo, J.; Gilmer, J.; Muelly, M.; Goodfellow, I.; Hardt, M.; Kim, B. Sanity Checks for Saliency Maps. In *Advances in Neural Information Processing Systems*; Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2018; Volume 31.
10. Guidotti, R.; Monreale, A.; Ruggieri, S.; Turini, F.; Giannotti, F.; Pedreschi, D. A survey of methods for explaining black box models. *ACM Comput. Surv. (CSUR)* **2018**, *51*, 1–42. [CrossRef]
11. Jacot, A.; Gabriel, F.; Hongler, C. Neural Tangent Kernel: Convergence and Generalization in Neural Networks. In *Advances in Neural Information Processing Systems*; Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2018; Volume 31.
12. Mei, S.; Montanari, A.; Nguyen, P.M. A mean field view of the landscape of two-layer neural networks. *Proc. Natl. Acad. Sci. USA* **2018**, *115*, E7665–E7671. [CrossRef]
13. Arora, S.; Du, S.; Hu, W.; Li, Z.; Wang, R. Fine-grained analysis of optimization and generalization for overparameterized two-layer neural networks. In Proceedings of the International Conference on Machine Learning, PMLR, Long Beach, CA, USA, 9–15 June 2019; pp. 322–332.
14. Cover, T.M.; Thomas, J.A. *Elements of Information Theory*; John Wiley & Sons: Hoboken, NJ, USA, 2012.
15. Huang, S.L.; Xu, X.; Zheng, L.; Wornell, G.W. An information theoretic interpretation to deep neural networks. In Proceedings of the 2019 IEEE International Symposium on Information Theory (ISIT), Paris, France, 7–12 July 2019; pp. 1984–1988.
16. Tishby, N.; Zaslavsky, N. Deep learning and the information bottleneck principle. In Proceedings of the Information Theory Workshop (ITW), Jerusalem, Israel, 26 April–1 May 2015; pp. 1–5.
17. Goldfeld, Z.; Polyanskiy, Y. The information bottleneck problem and its applications in machine learning. *IEEE J. Sel. Areas Inf. Theory* **2020**, *1*, 19–38. [CrossRef]
18. Huang, S.L.; Makur, A.; Zheng, L.; Wornell, G.W. An information-theoretic approach to universal feature selection in high-dimensional inference. In Proceedings of the 2017 IEEE International Symposium on Information Theory (ISIT), Aachen, Germany, 25–30 June 2017; pp. 1336–1340.
19. Arjovsky, M.; Chintala, S.; Bottou, L. Wasserstein generative adversarial networks. In Proceedings of the International Conference on Machine Learning, PMLR, Sydney, Australia, 6–11 August 2017; pp. 214–223.
20. Saxe, A.M.; Bansal, Y.; Dapello, J.; Advani, M.; Kolchinsky, A.; Tracey, B.D.; Cox, D.D. On the information bottleneck theory of deep learning. *J. Stat. Mech. Theory Exp.* **2019**, *2019*, 124020. [CrossRef]
21. Goodfellow, I.; Bengio, J.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2017.
22. Olga, R.; Jia, D.; Hao, S.; Jonathan, K.; Sanjeev, S.; Sean, M.; Zhiheng, H.; Andrej, K.; Aditya, K.; Michael, B.; et al. ImageNet Large Scale Visual Recognition Challenge. *Int. J. Comput. Vis.* **2015**, *115*, 211–252. [CrossRef]
23. Huang, S.L.; Zheng, L. Linear information coupling problems. In Proceedings of the 2012 IEEE International Symposium on Information Theory Proceedings, Cambridge, MA, USA, 1–6 July 2012; pp. 1029–1033.
24. Huang, S.L.; Makur, A.; Wornell, G.W.; Zheng, L. On universal features for high-dimensional learning and inference. *arXiv* **2019**, arXiv:1911.09105.
25. Hirschfeld, H.O. A connection between correlation and contingency. *Proc. Camb. Phil. Soc.* **1935**, *31*, 520–524. [CrossRef]
26. Gebelein, H. Das statistische problem der Korrelation als variations-und Eigenwertproblem und sein Zusammenhang mit der Ausgleichungsrechnung. *Z. Angew. Math. Mech.* **1941**, *21*, 364–379. [CrossRef]
27. Rényi, A. On Measures of Dependence. *Acta Math. Acad. Sci. Hung.* **1959**, *10*, 441–451. [CrossRef]

28. du Pin Calmon, F.; Makhdoumi, A.; Médard, M.; Varia, M.; Christiansen, M.; Duffy, K.R. Principal inertia components and applications. *IEEE Trans. Inf. Theory* **2017**, *63*, 5011–5038. [CrossRef]
29. Hsu, H.; Asoodeh, S.; Salamatian, S.; Calmon, F.P. Generalizing bottleneck problems. In Proceedings of the 2018 IEEE International Symposium on Information Theory (ISIT), Vail, CO, USA, 17–22 June 2018; pp. 531–535.
30. Hsu, H.; Salamatian, S.; Calmon, F.P. Correspondence analysis using neural networks. In Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics, PMLR, Okinawa, Japan, 16–18 April 2019; pp. 2671–2680.
31. Anantharam, V.; Gohari, A.; Kamath, S.; Nair, C. On hypercontractivity and a data processing inequality. In Proceedings of the 2014 IEEE International Symposium on Information Theory, Honolulu, HI, USA, 29 June–4 July 2014; pp. 3022–3026.
32. Raginsky, M. Strong data processing inequalities and Φ-Sobolev inequalities for discrete channels. *IEEE Trans. Inf. Theory* **2016**, *62*, 3355–3389. [CrossRef]
33. Polyanskiy, Y.; Wu, Y. Strong data-processing inequalities for channels and Bayesian networks. In *Convexity and Concentration*; Springer: Berlin/Heidelberg, Germany, 2017; pp. 211–249.
34. Greenacre, M.J. *Theory and Applications Of Correspondence Analysis*; Academic Press: London, UK, 1984.
35. Wang, H.; Vo, L.; Calmon, F.P.; Médard, M.; Duffy, K.R.; Varia, M. Privacy with estimation guarantees. *IEEE Trans. Inf. Theory* **2019**, *65*, 8025–8042. [CrossRef]
36. Breiman, L.; Friedman, J.H. Estimating Optimal Transformations for Multiple Regression and Correlation. *J. Am. Stat. Assoc.* **1985**, *80*, 614–619.
37. Sutskever, I.; Vinyals, O.; Le, Q.V. Sequence to sequence learning with neural networks. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2014; pp. 3104–3112.
38. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef]
39. Hastie, T.; Tibshirani, R.; Friedman, J. Neural Networks. In *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*; Springer: New York, NY, USA, 2009; pp. 389–416. [CrossRef]
40. Cybenko, G. Approximation by superpositions of a sigmoidal function. *Math. Control. Signals Syst.* **1989**, *2*, 303–314. [CrossRef]
41. Stoer, J.; Bulirsch, R. *Introduction to Numerical Analysis*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2013; Volume 12.
42. Alain, G.; Bengio, Y. Understanding intermediate layers using linear classifier probes. In Proceedings of the 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, 24–26 April 2017.
43. Sutskever, I.; Martens, J.; Dahl, G.; Hinton, G. On the importance of initialization and momentum in deep learning. In Proceedings of the International Conference on Machine Learning, PMLR, Atlanta, GA, USA, 17–19 June 2013; pp. 1139–1147.
44. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. In Proceedings of the 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, 7–9 May 2015; Bengio, Y., LeCun, Y., Eds.; Conference Track Proceedings.
45. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv* **2017**, arXiv:1704.04861.
46. Huang, G.; Liu, Z.; Weinberger, K.Q.; van der Maaten, L. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–29 July 2017; Volume 1, p. 3.
47. Chollet, F. Xception: Deep learning with depthwise separable convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–29 July 2017; pp. 1251–1258.
48. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the inception architecture for computer vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 2818–2826.
49. Szegedy, C.; Ioffe, S.; Vanhoucke, V.; Alemi, A.A. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. In Proceedings of the AAAI, San Francisco, CA, USA, 4–9 February 2017; pp. 4278–4284.
50. Xu, X.; Huang, S.L.; Zheng, L.; Zhang, L. The geometric structure of generalized softmax learning. In Proceedings of the 2018 IEEE Information Theory Workshop (ITW), Guangzhou, China, 25–29 November 2018; pp. 1–5.
51. Wen, W.; Wu, C.; Wang, Y.; Chen, Y.; Li, H. Learning structured sparsity in deep neural networks. *Adv. Neural Inf. Process. Syst.* **2016**, *29*, 2074–2082.
52. Wang, L.; Wu, J.; Huang, S.L.; Zheng, L.; Xu, X.; Zhang, L.; Huang, J. An efficient approach to informative feature extraction from multimodal data. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; Volume 33, pp. 5281–5288.
53. Lee, J.; Sattigeri, P.; Wornell, G. Learning new tricks from old dogs: Multi-source transfer learning from pre-trained networks. *Adv. Neural Inf. Process. Syst.* **2019**, *32*, 4370–4380.
54. Dembo, A.; Zeitouni, O. *Large Deviations Techniques and Applications*; Corrected Reprint of the Second (1998) Edition; Stochastic Modelling and Applied Probability; Springer: Berlin/Heidelberg, Germany, 2010; p. 38.
55. Sason, I.; Verdú, S. f-divergence Inequalities. *IEEE Trans. Inf. Theory* **2016**, *62*, 5973–6006. [CrossRef]
56. Eckart, C.; Young, G. The approximation of one matrix by another of lower rank. *Psychometrika* **1936**, *1*, 211–218. [CrossRef]