A Theoretical Insight into the Effect of Loss Function for Deep Semantic-Preserving Learning

Ali Akbari, *Member, IEEE,* Muhammad Awais, *Member, IEEE,* Manijeh Bashar, *Member, IEEE,*
and Josef Kittler, *Life Member, IEEE*

***Abstract*—Good generalisation performance is the fundamental goal of any machine learning algorithm. Using the uniform stability concept, this paper theoretically proves that the choice of loss function impacts on the generalisation performance of a trained deep neural network (DNN). The adopted stability based framework provides an effective tool for comparing the generalisation error bound with respect to the utilised loss function. The main result of our analysis is that using an effective loss function makes stochastic gradient descent more stable which consequently leads to the tighter generalisation error bound, and so better generalisation performance. To validate our analysis, we study learning problems in which the classes are semantically correlated. To capture this semantic similarity of neighbouring classes, we adopt the well-known semantics-preserving learning framework, namely label distribution learning (LDL). We propose two novel loss functions for the LDL framework and theoretically show that they provide stronger stability than the other widely used loss functions adopted for training DNNs. The experimental results on three applications with semantically correlated classes, including facial age estimation, head pose estimation and image aesthetic assessment, validate the theoretical insights gained by our analysis and demonstrate the usefulness of the proposed loss functions in practical applications.**

***Index Terms*—Generalisation performance, deep neural networks, loss function, statistical learning theory, semantic-preserving learning.**

I. Introduction

The fundamental goal of any machine learning approach is finding optimal solutions which generalise well from training data to unseen test data. In other words, a small gap, called *generalisation error*, between the performance on the training data and the test data is the fundamental objective of an arbitrary learning algorithm. Learning with powerful models such as deep neural networks (DNN) has achieved a step change in performance over recent years across a wide variety of tasks [2]–[7]. However, although learning algorithms, such as stochastic gradient descent (SGD), are able to recover solutions with small training error, understanding the generalisation performance is particularly critical for DNNs due to their well-known over-fitting issue.

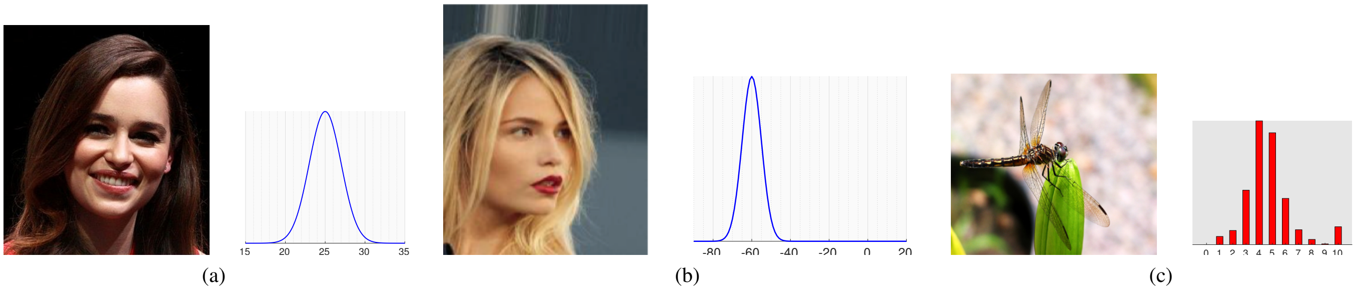
One way to assess the generalisation performance is to derive upper bounds for the generalisation error. This can be achieved using the notion of uniform stability of learning algorithms, introduced by Bousquet and Elisseeff [8], [9] for traditional learning algorithms and then developed by Hardt *et al.* [10] for randomised learning algorithms such as SGD. In fact, in training of DNNs with SGD, the distribution and the order of images in the training set introduce elements of randomness to the trained model. The uniform stability measures the sensitivity of the learnt solution to these perturbations in the training set. Intuitively, a good learning algorithm should be uniformly stable with respect to changes in the distribution of data and also the order of images in the training set. Hardt *et al.* [10] connect the concept of uniform stability to the generalisation error of a model trained by SGD. As a main consequence of this work, if the uniform stability holds for a learning algorithm (specifically SGD), then its generalisation error will be bounded with high probability [9].

As the current theoretical understanding of the effect of loss function on the generalisation performance is limited, it is not obvious how different loss functions affect the generalisation performance of an optimiser of the empirical risk minimisation, specifically the one found by SGD. As our main contribution in this paper, using statistical learning theory, we theoretically analyse the generalisation performance of a trained DNN with respect to the loss function adopted by the learning algorithm. We build our theoretical framework based on the uniform stability concept in [10] in order to compare the generalisation error bound of DNN models trained with SGD in conjunction with different loss functions. The main result of our analysis is that the tighter bound on the generalisation error can be obtained by utilising an effective and stable loss function for training DNNs via SGD. The proposed stability analysis method developed in this paper advances the theoretical understanding that could provide guidance for the selection of a loss function in practice. In essence, our result implies that the model trained using a stable loss function with moderate gradients generalises better.

We validate our theoretical analysis on machine learning problems in which there exists a semantic correlation among the adjacent classes. Considering these problems as standard classification problem, the correlation among the classes is ignored. Recently, Geng *et al.* [11] introduced a kind of a semantics-preserving learning framework, called *label distribution learning (LDL)*, to capture the semantic similarity information among classes during training. Given an input instance, the LDL framework allocates a descriptive degree to each label in the label set, which signifies the extent to which the label describes the instance. The true label of the instance has the highest degree. The complete set of all descriptive degrees of labels constitutes a *label distribution*

Ali Akbari, Muhammad Awais and Josef Kittler are with the Centre for Vision, Speech and Signal Processing (CVSSP), University of Surrey, Guildford, UK. e-mail: {ali.akbari,m.a.rana,m.bashar,j.kittler}@surrey.ac.uk.
Manijeh Bashar is with the Institute for Communication Systems (ICS), University of Surrey, Guildford, UK. (e-mail: m.bashar@surrey.ac.uk)

Parts of this work was presented at the ICML 2021 [1].



(a) (b) (c)

Fig. 1. The real-world applications of the LDL framework. (a) Facial age estimation – due to similarity between neighbouring ages, the label is a Gaussian distribution for a facial image at the age of 25. (b) Head pose estimation – since there exists similarity between faces with closed yaw angles, Gaussian distribution function is used to encode pose labels (shown image has yaw angle=$-60°$). (c) Image aesthetic assessment – the label is a mixture distribution for a given image, specified by crowd opinions, ranging from 0 to 10, on the aesthetic impact of the image.

over the label space. For instance, there is a strong correlation between the facial features of a person at a certain age and at the immediately preceding and subsequent ages [12], [13]. To capture this correlation, the LDL framework represents each age label as a discrete Gaussian distribution, called label distribution, which tries to introduce the cross-age correlations into the training phase. The expected value of each label distribution is set to equal the true age, and the variance of the label distributions corresponding to different ages is assumed to be the same. Fig. 1 shows three real-world applications which are tailor-made for being modelled by the LDL framework.

The well-known Kullback-Leibler (KL) divergence is currently employed as the loss function to measure the similarity between the predicted and the ground-truth label distributions for LDL [14]. As our second contribution in this paper, we propose two novel loss functions for the LDL framework which address the generalisation issue more effectively than the commonly used loss function, *i.e.* the KL divergence. We theoretically analyse the generalisation performance of the learnt DNN models using the proposed loss functions in comparison with the KL divergence. To this end, first, we derive upper bound on the values of the proposed loss functions and compare it with that of the KL divergence. Specifically, we prove that the proposed loss functions are upper bounded by the KL divergence. Moreover, we show that SGD using the proposed loss functions is generally more stable. These results are then used to compare the generalisation error bound of the DNN models trained by these loss functions. Finally, we experimentally demonstrate on three class-correlated applications, *i.e.* age estimation from a single face image of the subject, head pose estimation and image aesthetic assessment, that training DNNs with the proposed loss functions improves the generalisation performance compared to the DNN model trained by the KL divergence.

The rest of this paper is organised as follows: we briefly summarise the related work in Section II. We introduce the preliminaries and review the tools used to derive inequalities in Section III. Formulating the LDL framework, we introduce the proposed loss functions and compare their properties with those of the KL divergence in Section IV. The behaviour of the proposed loss functions is theoretically analysed in Section V. The theoretical findings are experimentally validated in Section VI. The last section concludes the paper.

Notation

The following notation is adopted in the rest of the paper. Uppercase and lowercase boldface letters are used for matrices and vectors, respectively. Scalars and sets are represented by standard and calligraphic fonts, respectively. The notation $\mathbb{E}\{\cdot\}$ and $\mathbb{P}\{\cdot\}$ denote expectation and probability, respectively. The random variables over which the expectation and probability are defined will be specified in subscript. Wherever it is clear from the text, we remove the subscripts for the sake of brevity. $|\cdot|$ and $\|\cdot\|$ stand for absolute value and the ℓ_2-norm of a vector. Moreover, $\mathbf{x}_i$ denotes the i-th element of a set.

II. Related Work

In order to design an efficient learning method [1], [12], [15]–[19], a deep understanding of the impact of various design choices on the generalisation performance is particularly critical. There is a venerable line of research focusing on insights into the generalisation performance of learning algorithms dating back more than thirty years [9], [20]–[22]. The existing strategies to improve the generalisation performance are drop-out [23], fast training [10], adversarial training [24], multitask learning [25], [26], dynamic learning rate and regularisation mechanisms [27], [28], neural network ensemble approach [29], designing residual based network architectures [30] and regularised algorithms which implement structured sparsity constraints [31], [32]. Although these approaches provide practical algorithms for improving the generalisation performance, there is still a lack of theoretical insight into how the above-mentioned method improve the generalisation performance.

To theoretically analyse the generalisation performance, one established approach in the literature is to derive upper bounds for the generalisation error. Focusing on the empirical error, there are different ways to establish upper bounds for the generalisation error, including stability [9], [10], [20], [33], [34], Vapnik-Chervonenkis (VC) dimension [33], [35], robustness [36] and PAC-Bayesian theory with the help of the properties of specific hypotheses [37]. The notion of uniform stability, which our work relies on, was introduced by Bousquet and Elisseeff [9] as a tool for deriving bounds for the generalisation error of deterministic learning algorithms. The follow-up works have since extended the notion of uniform stability to randomised algorithms such as bagging [8] and stochastic gradient descent [10]. According to these results, the optimiser of an empirical risk minimisation problem, implemented by the aforementioned learning algorithms, are uniformly stable under certain assumptions on regularity or convexity of the loss function.

The main outcome of the above-mentioned theoretical work is that the generalisation error is bounded by a vanishing function of the sample size. That means more training data leads to the tighter bound on the generalisation error. However, these work provide no guidance as to how to build a model with more generalisation capability where the the size of training data is fixed. Hardt *et al.* [10] also bounds the generalisation error of a model in terms of the number of iterations of SGD. Differs from these work, our focus in this paper is on theoretical insight into the effect of loss function on the generalisation performance of DNNs. To this end, we derive bounds on the generalisation error with respect to the properties of the employed loss function that induces stability. Rather than providing new insights into learning procedure or network architecture that yields the tighter bound on the generalisation error, our paper provides new theoretical understanding of how to design the loss function such that it improves the generalisation performance of DNNs trained by SGD.

III. Preliminaries

In this paper, the goal is to learn a model $f^{\theta}:\mathcal{X}\rightarrow\mathcal{Y}$, described by parameters $\theta\in\mathcal{H}$, between the input space $\mathcal{X}$ and its corresponding output space $\mathcal{Y}$. A common setting of such a learning algorithm is defined as

$$\underset{f^{\theta}\in\mathcal{F}}{\operatorname{argmin}}\,\mathbb{E}_{\mathbf{z}\sim\mathcal{D}}[\ell(f^{\theta};\mathbf{z})], \tag{1}$$

where it seeks a model f^{θ} over some hypothesis space $\mathcal{F}$ that minimises the true (expected) risk $R_{\text{true}}(f^{\theta})\triangleq\mathbb{E}_{\mathbf{z}\sim\mathcal{D}}[\ell(f^{\theta};\mathbf{z})]$ with respect to sample $\mathbf{z}=(\mathbf{x},\mathbf{y})\in\mathcal{X}\times\mathcal{Y}$, drawn according to an unknown distribution $\mathcal{D}$. $\ell:\mathcal{Y}\times\mathcal{Y}\rightarrow\mathbb{R}^{+}$ is the loss function which measures the accuracy of a hypothesis f^{θ} based on the discrepancy between the predicted and real outputs. We also sometime write $\ell(f^{\theta};\mathbf{z})=\ell(f^{\theta}(\mathbf{x});\mathbf{y})$ instead.

Since the distribution $\mathcal{D}$ is unknown, the minimisation problem (1) cannot be solved directly. Instead, the true risk $R_{\text{true}}(f^{\theta})$ is estimated with the empirical risk over a training set. Consider a finite set of N training samples $\mathcal{S}=\{\mathbf{z}_i, i=1,2,\cdots,N\}$, where $\mathbf{z}_i=(\mathbf{x}_i,\mathbf{y}_i)$, i.i.d. sampled according to an unknown distribution $\mathcal{D}$. The empirical risk is then defined as $R_{\text{emp}}(f^{\theta})\triangleq\frac{1}{N}\sum_{i=1}^{N}\ell(f^{\theta};\mathbf{z}_i)$.

A learning algorithm $\mathcal{A}:(\mathcal{X}\times\mathcal{Y})^{N}\rightarrow\mathcal{Y}^{\mathcal{X}}$ is used to solve the minimisation problem (1). In the context of neural networks, SGD is the learning algorithm $\mathcal{A}$ for dealing with such an optimisation problem. SGD is a randomised algorithm. The randomness of SGD algorithm appears either in the initialisation procedure by which the network's weights are randomly initialised or in the random selection of training samples $\mathcal{S}$ during the network's update. To simplify the notation, throughout this paper, we consider the random choice of one sample from the training set $\mathcal{S}$ at each iteration to indicate the only nature of randomness of the learning algorithm $\mathcal{A}$. To model this randomness of SGD algorithm, let set $\mathcal{R}=\{r_1\cdots r_T\}$ denote the set of random indices of samples in the training set $\mathcal{S}$. We denote $f^{\theta}_{\mathcal{S},\mathcal{R}}$ as the output of the algorithm $\mathcal{A}$ applied to a training dataset $\mathcal{S}$, with a random set $\mathcal{R}$. Note that $R_{\text{true}}(f^{\theta}_{\mathcal{S},\mathcal{R}})$ and $R_{\text{emp}}(f^{\theta}_{\mathcal{S},\mathcal{R}})$ are random variables depending on $\mathcal{S}$ and $\mathcal{R}$.

The essential task of a learning algorithm, such as SGD algorithm, is to find a good hypothesis f^{θ} w.r.t. a suitably chosen loss function ℓ such that the difference between the performance of the learnt model $f^{\theta}_{\mathcal{S},\mathcal{R}}$ over the training set $\mathcal{S}$ and any other test set endowed with an unknown distribution $\mathcal{D}$ is minimised. One way to evaluate the efficiency of the learning algorithm is to derive upper bounds for the *generalisation error* as a measure of the performance of the learning algorithm.

Definition 1 (Generalisation Error)**.** *Given a training set $\mathcal{S}$, the generalisation error of the output model $f^{\theta}_{\mathcal{S},\mathcal{R}}$, trained using the learning algorithm $\mathcal{A}$ on $\mathcal{S}$ with a set of random indices $\mathcal{R}$, is the difference between the empirical and true risk, i.e. $E(\mathcal{S},\mathcal{R})=R_{\text{true}}(f^{\theta}_{\mathcal{S},\mathcal{R}})-R_{\text{emp}}(f^{\theta}_{\mathcal{S},\mathcal{R}})$. Note that $E(\mathcal{S},\mathcal{R})$ is a random variable depending on $\mathcal{S}$ and $\mathcal{R}$.*

A manifestation of the lack of generalisation is also called *over-fitting*. If an algorithm has the tighter bound, its generalisation performance is expected to be better. The study we describe here intends to compare the upper bounds on the generalisation error of the obtained model $f^{\theta}_{\mathcal{S},\mathcal{R}}$ using the learning algorithm $\mathcal{A}$ with respect to different loss functions. In our analysis, we will use the notion of uniform stability [9] to uncover the link between stability and generalisation performance of SGD algorithm using different loss functions. Our analysis concerns the LDL based formulation being appropriate for the tasks with strong correlation among the classes. For brevity, in the following, $f_{\mathcal{S},\mathcal{R}}$, $R_{\text{true}}(f_{\mathcal{S},\mathcal{R}})$ and $R_{\text{emp}}(f_{\mathcal{S},\mathcal{R}})$ are sometimes used as shorthand for $f^{\theta}_{\mathcal{S},\mathcal{R}}$, $R_{\text{true}}(f^{\theta}_{\mathcal{S},\mathcal{R}})$ and $R_{\text{emp}}(f^{\theta}_{\mathcal{S},\mathcal{R}})$ if their meaning is clear from the context.

A. Bounded difference inequality

In our analysis, the bounded difference inequality (BDI), proved by McDiarmid [38], is central to linking the uniform stability and generalisation. Let $\mathcal{Z}$ be some set and $G:\mathcal{Z}^{n}\rightarrow\mathbb{R}$ be any measurable function. Consider two sets $\mathcal{Q},\mathcal{Q}'\in\mathcal{Z}^{n}$, such that $\mathcal{Q}$ and $\mathcal{Q}'$ differ in at most one element. If there exists constant ρ such that

$$\sup_{\mathcal{Q},\mathcal{Q}'\in\mathcal{Z}^{n}}|G(\mathcal{Q})-G(\mathcal{Q}')|\leq\rho, \tag{2}$$

then $\forall\epsilon>0$

$$\mathbb{P}_{\mathcal{Q}}\big[G(\mathcal{Q})-\mathbb{E}_{\mathcal{Q}}[G(\mathcal{Q})]\geq\epsilon\big]\leq\exp(-2\epsilon^{2}/n\rho^{2}). \tag{3}$$

The inequality (2) is called *bounded difference condition*. Intuitively, $G(\cdot)$ satisfies the bounded differences property (2) if changing only one element of $\mathcal{Q}$ at a time cannot make $G(\cdot)$ deviates too far. It should not be too surprising that these types of functions thus concentrate somewhat strongly around their average, and this intuition is made precise by Eq. (3).

B. Semantics-Preserving Learning

Let $(\mathbf{x}, y)$ denote a training sample, where $\mathbf{x}$ and y represent the input instance and its corresponding class label, respectively. The class label y is a scalar value from the set of possible age labels $\mathcal{L} = \{l_{min}, \cdots, l_{max}\}$. Here, the goal is to learn a mapping function between the input instance $\mathbf{x}$ and its corresponding label y. A typical learning algorithm models this problem as *general classification* problem and uses one-hot encoding for representing the labels. In this type of modelling, label y is encoded by a binary vector $\mathbf{y} \in \mathbb{R}^K$, whose n-th element is one if the input sample $\mathbf{x}$ belongs to the n-th label in $\mathcal{L}$.

In the one-hot label modelling, it is assumed the classes are uncorrelated rather than dependent. However, this is a strong assumption. In some real-world application, there exists semantic correlation among classes. For instance, in age estimation from the face appearance, close classes (ages) are correlated due to the fact that the images of faces of close age labels usually share some visual features. Therefore, we need to build a semantics-preserving learning framework for this kind of learning problems. Instead of using the one-hot label encoding, recently, Geng *et al.* [11] model this correlation among neighbouring classes by assigning a label distribution $\mathbf{y} = [y_1, y_2, \cdots, y_K] \in \mathbb{R}^K$ to each input sample $\mathbf{x}$, where $\forall y_i, 0 \leq y_i \leq 1$ and $\sum_{i=1}^{K} y_i = 1$. For instance, in this type of modelling for the age estimation problem, the label vector $\mathbf{y}$ is assumed to have a Gaussian distribution, centred at the true age y with a standard deviation, σ (see Fig. 1 for other applications.) With this kind of label modelling, a semantic-preserving learning framework, namely *label distribution learning* problem, is built.

IV. Loss Functions

To optimise the parameters of a DNN model so that the model describes the label distributions well, we need to choose an appropriate loss function to accurately measure the meaningful distance between the predicted and ground-truth label distributions to work with the adopted learning algorithm. Given a typical face sample $\mathbf{x}$, let $\mathbf{y}$ and $\hat{\mathbf{y}} = f^{\theta}(\mathbf{x})$ denote the corresponding ground-truth label distribution and output label distribution estimated by the DNN, respectively. Further, consider y_k and $\hat{y}_k$ as the k-element of $\mathbf{y}$ and $\hat{\mathbf{y}}$, respectively. In the existing LDL based methods, the KL divergence is widely employed as the loss function to measure the similarity between the estimated label distribution and ground-truth [14]. The KL loss function is defined as

$$\ell_{KL}(\hat{\mathbf{y}}, \mathbf{y}) = \sum_{k=1}^{K} y_k \log(\frac{y_k}{\hat{y}_k}). \tag{4}$$

However, the KL divergence, due to some well-known limitations including being unbounded and asymmetric [3], [12], is unable to accurately measure the distance between two distributions. Different from the exiting work, we propose two novel loss functions for the LDL frameworks which address the aforementioned issues associated with the KL divergence and can handle the LDL problem more effectively.

We propose the use of symmetric version of the KL divergence, *i.e.* Jensen-Shannon divergence (JS) as the loss function for the LDL framework. The JS loss function is defined as

$$\ell_{JS}(\hat{\mathbf{y}}, \mathbf{y}) = \frac{1}{2} \sum_{k=1}^{K} y_k \log(\frac{y_k}{\tilde{y}_k}) + \hat{y}_k \log(\frac{\hat{y}_k}{\tilde{y}}), \tag{5}$$

where $\tilde{y} = (\hat{y}_k + y_k)/2$. Further, we consider a novel parametric loss function for the LDL framework via generalising the Jeffries-Matusita distance [12], [39]. We call this loss as Generalised Jeffries-Matusita distance (GJM). The GJM loss function is defined as

$$\ell_{GJM}(\hat{\mathbf{y}}, \mathbf{y}) = \sum_{k=1}^{K} |y_k^{\alpha} - \hat{y}_k^{\alpha}|^{\frac{1}{\alpha}} = \sum_{k=1}^{K} y_k \left| 1 - \left(\frac{\hat{y}_k}{y_k}\right)^{\alpha} \right|^{\frac{1}{\alpha}}, \tag{6}$$

where α is in the range $(0, 1]$. In our experiments, we found the best performance is obtained when $0.3 \leq \alpha \leq 0.6$. In the rest of the paper, we consider α as 0.5, unless stated otherwise.

In the following sections, our generalisation error analysis with respect to these loss functions will be presented. We will theoretically prove the superiority of the GJM measure in terms of generalisation, and experimentally confirm the theoretical finding by measuring its performance in comparison with the two other measures, *i.e.* KL and JS divergences. Throughout this paper, the architecture of the network is the same.

Loss Function Properties

The following definitions and theorems provide the foundation for our analysis of how the loss function impacts on the generalisation ability of the trained model, which will be discussed in the next section. In our analysis, a loss function needs to satisfy the following properties:

Definition 2 (Lipschitz property). *A loss function $\ell(\hat{\mathbf{y}}, \mathbf{y})$ is γ-Lipschitz (admissible) with respect to the output vector $\hat{\mathbf{y}}$, if for $\gamma \geq 0$ and $\forall \mathbf{u}, \mathbf{v} \in \mathbb{R}^K$ we have*

$$|\ell(\mathbf{u}, \mathbf{y}) - \ell(\mathbf{v}, \mathbf{y})| \leq \gamma \|\mathbf{u} - \mathbf{v}\|. \tag{7}$$

We use $\|\cdot\|$ to denote the ℓ_2-norm of vectors. Intuitively, a Lipschitz function is bounded in terms of how fast it is allowed to change.

Definition 3 (Smoothness). *A loss function $\ell(\hat{\mathbf{y}}, \mathbf{y})$ is η-smooth with respect to the prediction output vector $\hat{\mathbf{y}}$, if its gradient $\nabla \ell(\hat{\mathbf{y}}, \mathbf{y})$ is η-Lipschitz, that is for $\eta \geq 0$ and $\forall \mathbf{u}, \mathbf{v} \in \mathbb{R}^K$ we have*

$$\|\nabla \ell(\mathbf{u}, \mathbf{y}) - \nabla \ell(\mathbf{v}, \mathbf{y})\| \leq \eta \|\mathbf{u} - \mathbf{v}\|. \tag{8}$$

Intuitively, the curvature of the loss function is bounded by the property of the η-smoothness.

Theorem 1. *Let function $h : (0, \infty) \rightarrow \mathbb{R}$ be convex, such that $h(1) = 0$. Let us define*

$$I(\hat{\mathbf{y}}, \mathbf{y}) = \sum_{k=1}^{K} y_k h\left(\frac{\hat{y}_k}{y_k}\right), \tag{9}$$

as a distance function. If $h(\cdot)$ is γ-Lipschitz, i.e.

$$|h(x) - h(z)| \leq \gamma |x - z| \quad \forall x, z, \tag{10}$$

then $I(\hat{\mathbf{y}}, \mathbf{y})$ is also γ-Lipschitz. Further, since $h(\cdot)$ is convex, $I(\hat{\mathbf{y}}, \mathbf{y})$ is also convex with respect to its first argument.

Proof. Let $x = \frac{u_k}{y_k}$ and $z = \frac{v_k}{y_k}$. Then, from (10), we have

$$\left| h\left(\frac{u_k}{y_k}\right) - h\left(\frac{v_k}{y_k}\right) \right| \leq \gamma \left| \frac{u_k}{y_k} - \frac{v_k}{y_k} \right| \quad \forall k \in \{1, \cdots L\}. \tag{11}$$

Multiplying both sides of (11) by y_k, and then employing summation on all the obtained inequalities for all $k \in \{1, \cdots L\}$, we obtain

$$\sum_{k=1}^{K} \left| y_k h\left(\frac{u_k}{y_k}\right) - y_k h\left(\frac{v_k}{y_k}\right) \right| \leq \gamma \sum_{k=1}^{K} |u_k - v_k| . \tag{12}$$

Using the generalised triangle inequality, we get

$$\begin{aligned} &\left| \sum_{k=1}^{K} y_k h\left(\frac{u_k}{y_k}\right) - \sum_{k=1}^{K} y_k h\left(\frac{v_k}{y_k}\right) \right| \\ &\leq \sum_{k=1}^{K} \left| y_k h\left(\frac{u_k}{y_k}\right) - y_k h\left(\frac{v_k}{y_k}\right) \right| \leq \gamma \sum_{k=1}^{K} |u_k - v_k| . \end{aligned} \tag{13}$$

Finally, we obtain

$$|I(\mathbf{u}, \mathbf{y}) - I(\mathbf{v}, \mathbf{y})| \leq \gamma \|\mathbf{u} - \mathbf{v}\|. \tag{14}$$

and the Lipschitz property of $I(\hat{\mathbf{y}}, \mathbf{y})$ is proved.

We now prove that the convexity of $h(\cdot)$ implies the convexity of $I(\hat{\mathbf{y}}, \mathbf{y})$ with respect to its first argument. Let $\mathbf{u}, \mathbf{v} \in \mathcal{Y}$ be two probability distributions with all values nonzero and $t \in [0.1]$. Then we have

$$I(t\mathbf{u} + (1-t)\mathbf{u}, \mathbf{y}) = \sum_{k=1}^{K} y_k h\left(\frac{tu_k + (1-t)v_k}{y_k}\right). \tag{15}$$

Due to the convexity of h, we have $\forall k \in \{1, \cdots, K\}$

$$h\left(\frac{tu_k + (1-t)v_k}{y_k}\right) \leq th\left(\frac{u_k}{y_k}\right) + (1-t)h\left(\frac{v_k}{y_k}\right). \tag{16}$$

Summing over k from 1 to K and utilizing (15) we get

$$I(t\mathbf{u} + (1-t)\mathbf{u}, \mathbf{y}) \leq tI(\mathbf{u}, \mathbf{y}) + (1-t)I(\mathbf{v}, \mathbf{y}) \tag{17}$$

proving the desired result. $\blacksquare$

Remark. With $h_{KL}(x) = -\log(x), x > 0$ and $h_{JS}(x) = x\log(\frac{2x}{x+1}) + \log(\frac{2}{x+1})$, then obviously $\ell_{KL}(\hat{\mathbf{y}}, \mathbf{y}) = I(\hat{\mathbf{y}}, \mathbf{y})$ and $\ell_{JS}(\hat{\mathbf{y}}, \mathbf{y}) = I(\hat{\mathbf{y}}, \mathbf{y})$, respectively. It is also obvious that if $h_{GJM}(x) = |1 - x^{\alpha}|^{\frac{1}{\alpha}}, x > 0$, then $\ell_{GJM}(\hat{\mathbf{y}}, \mathbf{y}) = I(\hat{\mathbf{y}}, \mathbf{y})$. Consequently, since $h_{KL}(x), h_{JS}(x)$ and $h_{GJM}(x)$ are convex functions, then $\ell_{KL}(\hat{\mathbf{y}}, \mathbf{y}), \ell_{JS}(\hat{\mathbf{y}}, \mathbf{y})$ and $\ell_{GJM}(\hat{\mathbf{y}}, \mathbf{y})$ are also convex.

Lemma 1. *A function $h : (0, \infty) \to \mathbb{R}$ is γ-Lipschitz, if γ satisfies*

$$\gamma = \sup_{x} |h'(x)|. \tag{18}$$

That means the value of γ must equal the maximum value $|h'(x)|$ can assume.

Proof. This lemma can be easily derived from the definition of Lipschitz property. $\blacksquare$

The practical result is embodied in the following corollary, which is a prerequisite for our analysis in the next section.

Corollary 1. *Given that the GJM, JS and KL loss functions are γ_{GJM}-Lipschitz, γ_{JS}-Lipschitz and γ_{KL}-Lipschitz, respectively, the following inequality holds:*

$$\begin{aligned} \gamma_{GJM} &\leq \gamma_{KL} \\ \gamma_{JS} &\leq \gamma_{KL} \end{aligned} \tag{19}$$

Proof. As $h_{KL}(x) = -\log(x), x > 0$ and $h_{JS}(x) = x\log(\frac{2x}{x+1}) + \log(\frac{2}{x+1})$, then obviously $\ell_{KL}(\hat{\mathbf{y}}, \mathbf{y}) = I(\hat{\mathbf{y}}, \mathbf{y})$ and $\ell_{JS}(\hat{\mathbf{y}}, \mathbf{y}) = I(\hat{\mathbf{y}}, \mathbf{y})$, respectively. It is also obvious that if $h_{GJM}(x) = |1 - x^{\alpha}|^{\frac{1}{\alpha}}, x > 0$, then $\ell_{GJM}(\hat{\mathbf{y}}, \mathbf{y}) = I(\hat{\mathbf{y}}, \mathbf{y})$. Then, we have

$$\begin{aligned} h'_{KL}(x) &= -\frac{1}{x}, \\ h'_{JS}(x) &= \log\left(\frac{2x}{x+1}\right) - \frac{1}{x+1}, \\ h'_{GJM}(x) &= \text{sign}(x^{\alpha} - 1)x^{\alpha-1}|x^{\alpha} - 1|^{\frac{1-\alpha}{\alpha}}. \end{aligned} \tag{20}$$

Figure 2 shows the absolute value of the derivative of the KL, JS and GJM loss functions as a function of x. As can be seen $|h'_{GJM}(x)|$ is close to but smaller than $|h'_{JS}(x)|$ and both $|h'_{GJM}(x)|$ and $|h'_{JS}(x)|$ are smaller than $|h'_{KL}(x)|$. From Lemma 1, this implies the inequality in (19) holds. We further provide a proof for above-mentioned observation. First, we prove that $|h'_{JS}(x)| \leq |h'_{KL}(x)|$, *i.e.*

$$\left| \frac{1}{x+1} - \log\left(\frac{2x}{x+1}\right) \right| \leq \left| \frac{1}{x} \right|. \tag{21}$$

First, given the fact that $\left|\frac{1}{x+1}\right| \leq \left|\frac{1}{x}\right|, \forall x$. Then, exploiting the logarithmic inequality $\log(x) \leq x - 1$ for $x \geq 0$, after some mathematical simplification, we have

$$\begin{aligned} \left| \frac{1}{x+1} - \log\left(\frac{2x}{x+1}\right) \right| &\leq \left| \frac{1}{x+1} - (\log(2) + x - 1 - x) \right| \\ &\leq \left| \frac{1}{x+1} \right| < \left| \frac{1}{x} \right|. \end{aligned} \tag{22}$$

This implies the inequality (21). Next, we aim to prove that $|h'_{GJM}(x)| \leq |h'_{KL}(x)|$, *i.e.*

$$\left| 1 - \frac{1}{\sqrt{x}} \right| \leq \left| \frac{1}{x} \right|. \tag{23}$$

Equation (23) is equivalent to $|x - \sqrt{x}| \leq 1$, which results in $x \leq 2.7$ after some mathematical simplification. Note that we experimentally found that the variable x always satisfies the above condition.

$\blacksquare$

As the last theorem in this section, we now provide upper bounds for the above-mentioned loss functions.

Theorem 2. *For two distribution $\mathbf{y}, \hat{\mathbf{y}} \in \mathbb{R}^K$, the KL and JS loss functions provide upper bounds on the GJM loss function with the parameter $\alpha = 0.5$, i.e. we have:*

$$\begin{aligned} \ell_{GJM}(\hat{\mathbf{y}}, \mathbf{y}) &\leq \ell_{KL}(\hat{\mathbf{y}}, \mathbf{y}) \\ \ell_{GJM}(\hat{\mathbf{y}}, \mathbf{y}) &\leq \ell_{JS}(\hat{\mathbf{y}}, \mathbf{y}). \end{aligned} \tag{24}$$

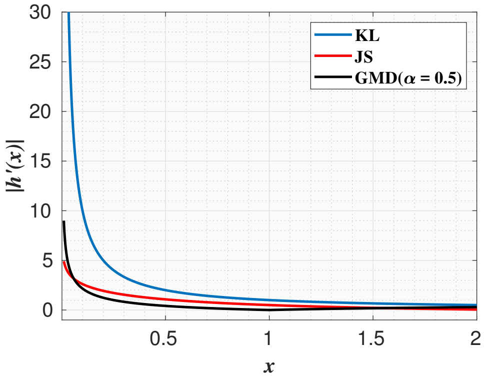


Fig. 2. Absolute value of derivative of loss functions at different points x.

Proof. First, we prove the first inequality. The use of binomial theorem and Jensen's inequality gives

$$\begin{aligned}\ell_{GJM}(\hat{\mathbf{y}},\mathbf{y}) &= \sum_{k=1}^{K} y_k \left|1-\left(\frac{\hat{y}_k}{y_k}\right)^{\alpha}\right|^{\frac{1}{\alpha}}\\ &\overset{(a)}{=} \sum_{k=1}^{K} y_k \sum_{i=0}^{1/\alpha} (-1)^i \binom{1/\alpha}{i} \left(\frac{\hat{y}_k}{y_k}\right)^{i\alpha}\\ &= \sum_{k=1}^{K} y_k \sum_{i=0}^{1/\alpha} (-1)^i \binom{1/\alpha}{i} \exp\left(i\alpha \log\left(\frac{\hat{y}_k}{y_k}\right)\right)\\ &\overset{(b)}{\leq} \sum_{i=0}^{1/\alpha} (-1)^i \binom{1/\alpha}{i} \exp\left(i\alpha \sum_{k=1}^{K} y_k \log\left(\frac{\hat{y}_k}{y_k}\right)\right)\\ &= \sum_{i=0}^{1/\alpha} (-1)^i \binom{1/\alpha}{i} \exp(-i\alpha\ell_{KL}(\hat{\mathbf{y}},\mathbf{y}))\\ &= (1-\exp(-\alpha\ell_{KL}(\hat{\mathbf{y}},\mathbf{y})))^{\frac{1}{\alpha}},\end{aligned} \tag{25}$$

where inequalities a and b are due to the Binomial theorem and the Jensen's inequality, respectively. Note that $\ell_{KL}(\hat{\mathbf{y}},\mathbf{y}) \in [0,\infty)$ [40]. The use of Bernoulli's inequality $\exp(x) \geq 1+x$ gives

$$\begin{aligned}\ell_{GJM}(\hat{\mathbf{y}},\mathbf{y}) &\leq (1-\exp(-\alpha\ell_{KL}(\hat{\mathbf{y}},\mathbf{y})))^{\frac{1}{\alpha}}\\ &\leq 1-\exp(-\alpha\ell_{KL}(\hat{\mathbf{y}},\mathbf{y})) \leq \ell_{KL}(\hat{\mathbf{y}},\mathbf{y}),\end{aligned} \tag{26}$$

and the first inequality is proved. The proof of the second inequality can be inferred by combining the inequalities in [41, p. 48] and [42, Theorem 3.2]. ■

V. Stability and Generalisation Error Bound

In this section, we analyse the generalisation error of DNNs trained with SGD with respect to different loss functions using the notion of stability as a tool. As mentioned in Section III, our approach to upper bounding the generalisation error E is based on the concept of uniform stability, introduced in [9], [10]. We follow the notion of stability introduced by Hardt *et al.* [10] with respect to randomness of the learning algorithm $\mathcal{A}$. Here, stability refers to the stability of the hypothesis at the output of the algorithm with respect to small changes of its input.

Definition 4 (Uniform Stability)**.** *Let set $\mathcal{S}'$ denote a training set of the same size as set $\mathcal{S}$ drawn according to an unknown distribution $\mathcal{D}$, such that $\mathcal{S}$ and $\mathcal{S}'$ differ in only one element. Let $f_{\mathcal{S},\mathcal{R}}$ and $f_{\mathcal{S}',\mathcal{R}}$ denote the optimal neural models obtained by the learning algorithm $\mathcal{A}$ with a set of random parameters $\mathcal{R}$ over the training sets $\mathcal{S}$ and $\mathcal{S}'$, respectively. The learning algorithm $\mathcal{A}$ is β-uniformly stable with respect to a specific loss function ℓ, if the following condition holds:*

$$\forall\ \mathcal{S},\ \mathcal{S}' \quad \sup_{\mathbf{z}} \mathbb{E}_{\mathcal{R}}\left[\,|\ell(f_{\mathcal{S},\mathcal{R}};\mathbf{z}) - \ell(f_{\mathcal{S}',\mathcal{R}};\mathbf{z})|\,\right] \leq \beta, \tag{27}$$

where the expectation is taken over randomness of the learning algorithm $\mathcal{A}$ which appears in the random selection of training samples $\mathcal{S}$ during the network's update.

From the above definition, if a given learning algorithm is β-uniformly stable, it has the property that changing one point in the training set and keeping others fixed leads to at most β-change in the error of the produced model with any random permutation of training samples in $\mathcal{S}$.

As proved by Bousquet and Elisseeff [9], if a deterministic learning algorithm is β-uniformly stable, its generalisation error is also upper bounded by a factor of β. This implies the following: If a learning algorithm with a specific loss function satisfies the stability condition with a more restrictive stability measure, the tighter generalisation error bound may be expected. However, these results concern only deterministic learning algorithms. Different from Bousquet's stability definition, our notion of stability concerns the randomness of the learning algorithm, similar to works presented in [10], [20]. Based on this, we present a new result which uncovers the relation between stability and generalisation of randomised learning algorithms, being suitable for analysing the performance of neural networks with respect to employed loss function. This assertion is reformulated in the following theorem.

Theorem 3. *Consider a loss function ℓ whose value ranges in $[0,L]$. Let $f_{\mathcal{S},\mathcal{R}}$ denote the optimal neural model obtained by the learning algorithm $\mathcal{A}$ with the set of random parameters $\mathcal{R}$ over the training sets $\mathcal{S}$. Let $\mathcal{A}$ be β-uniformly stable with respect to the loss function ℓ. Further, assume there exists a constant ρ for which the loss function $\ell(f_{\mathcal{S},\mathcal{R}},\mathbf{z})$ satisfies the bounded difference condition* (2) *with respect to the set of random parameters $\mathcal{R}$. Then, for any random draw of $\mathcal{S}$ and $\mathcal{R}$, the following bound holds with probability at least $1-\delta$:*

$$\begin{aligned}&R_{\text{true}}(f_{\mathcal{S},\mathcal{R}}) - R_{\text{emp}}(f_{\mathcal{S},\mathcal{R}}) \leq\\ &\rho\sqrt{T\log(2/\delta)} + \beta\left(1+\sqrt{2N\log(2/\delta)}\right) + L\sqrt{\frac{\log(2/\delta)}{2N}}.\end{aligned} \tag{28}$$

Proof. Let $E(\mathcal{S},\mathcal{R}) = R_{\text{true}}(f_{\mathcal{S},\mathcal{R}}) - R_{\text{emp}}(f_{\mathcal{S},\mathcal{R}})$, be a random variable depending on $\mathcal{S}$ and $\mathcal{R}$. Let $f_{\mathcal{S},\mathcal{R}}$ and $f_{\mathcal{S},\mathcal{R}'}$ be two output models using the learning algorithm $\mathcal{A}$ applied on the training set $\mathcal{S}$ with the two sets of random parameters $\mathcal{R}$ and $\mathcal{R}'$, respectively. We apply the BDI (3) by considering function G and set Q as $\ell(f_{\mathcal{S},\mathcal{R}};\mathbf{z})$ and $\mathcal{R}$, respectively. Assume $\mathcal{R}$ and $\mathcal{R}'$ differ only in two elements[1]. Note the BDI cannot be

[1] Recall that $\mathcal{R}$ is a set of random indices in $\mathcal{S}$. So, if $\mathcal{R}$ and $\mathcal{R}'$ differ in one element, unavoidably $\mathcal{R}$ and $\mathcal{R}'$ would differ in another element as well.

applied directly in this case. So we partition each $\mathcal{R}$ and $\mathcal{R}'$ in two subsets $\mathcal{R}_1, \mathcal{R}_2$ and $\mathcal{R}'_1, \mathcal{R}'_2$ such that the corresponding subsets, *i.e.* $\mathcal{R}_1, \mathcal{R}'_1$ and $\mathcal{R}_2, \mathcal{R}'_2$ differ only in one element. Using the bounded difference conditions (2), there would be a constant $\rho = \max(\rho_1, \rho_2)$ such that for every $\mathbf{z}$ and $\mathcal{S}$, we have the following bounded difference conditions with respect to ℓ:

$$\sup_{\mathcal{R}_1, \mathcal{R}'_1} \left|\ell(f_{\mathcal{S},\mathcal{R}_1}; \mathbf{z}) - \ell(f_{\mathcal{S},\mathcal{R}'_1}; \mathbf{z})\right| \leq \rho \tag{29}$$

and

$$\sup_{\mathcal{R}_2, \mathcal{R}'_2} \left|\ell(f_{\mathcal{S},\mathcal{R}_2}; \mathbf{z}) - \ell(f_{\mathcal{S},\mathcal{R}'_2}; \mathbf{z})\right| \leq \rho, \tag{30}$$

Then, for every $S, \mathcal{R}_1, \mathcal{R}'_1$, we have

$$\begin{aligned}
&|E(\mathcal{S}, \mathcal{R}_1) - E(\mathcal{S}, \mathcal{R}'_1)| \\
&\quad = \Big|\mathbb{E}_{\mathbf{z}\sim\mathcal{D}}\big[\ell(f_{\mathcal{S},\mathcal{R}_1}; \mathbf{z})\big] - \mathbb{E}_{\mathbf{z}\sim\mathcal{D}}\big[\ell(f_{\mathcal{S},\mathcal{R}'_1}; \mathbf{z})\big] \\
&\qquad\qquad - \frac{1}{N}\sum_{i=1}^{N} \ell(f_{\mathcal{S},\mathcal{R}_1}; \mathbf{z}_i) + \frac{1}{N}\sum_{i=1}^{N} \ell(f_{\mathcal{S},\mathcal{R}'_1}; \mathbf{z}_i)\Bigg| \\
&\quad \leq \mathbb{E}_{\mathbf{z}\sim\mathcal{D}}\Bigg[\left|\ell(f_{\mathcal{S},\mathcal{R}_1}; \mathbf{z}) - \ell(f_{\mathcal{S},\mathcal{R}'_1}; \mathbf{z})\right|\Bigg] \\
&\qquad\qquad + \frac{1}{N}\sum_{i=1}^{N} \left|\ell(f_{\mathcal{S},\mathcal{R}_1}; \mathbf{z}_i) - \ell(f_{\mathcal{S},\mathcal{R}'_1}; \mathbf{z}_i)\right| \\
&\quad \leq 2\rho.
\end{aligned} \tag{31}$$

Applying the BDI (3) results in the following inequality

$$\mathbb{P}_{\mathcal{R}_1}\big[E(\mathcal{S}, \mathcal{R}_1) - \mathbb{E}_{\mathcal{R}_1}\left[E(\mathcal{S}, \mathcal{R}_1)\right] \geq \epsilon\big] \leq \exp(-\epsilon^2/2T\rho^2). \tag{32}$$

Following the same lines for $\mathcal{R}_2, \mathcal{R}'_2$, the following inequality also holds

$$\mathbb{P}_{\mathcal{R}_2}\big[E(\mathcal{S}, \mathcal{R}_2) - \mathbb{E}_{\mathcal{R}_2}\left[E(\mathcal{S}, \mathcal{R}_2)\right] \geq \epsilon\big] \leq \exp(-\epsilon^2/2T\rho^2). \tag{33}$$

By combining the above two inequalities, we obtain:

$$\mathbb{P}_{\mathcal{R}}\big[E(\mathcal{S}, \mathcal{R}) - \mathbb{E}_{\mathcal{R}}\left[E(\mathcal{S}, \mathcal{R})\right] \geq \epsilon\big] \leq \exp(-\epsilon^2/T\rho^2). \tag{34}$$

By setting the r.h.s. equal to ν, the following inequality holds with probability at least $1 - \nu$:

$$E(\mathcal{S}, \mathcal{R}) \leq \rho\sqrt{T\log(1/\nu)} + \mathbb{E}_{\mathcal{R}}\left[E(\mathcal{S}, \mathcal{R})\right]. \tag{35}$$

To bound the random variable $E(\mathcal{S}, \mathcal{R})$, we now provide the upper bound for $\mathbb{E}_{\mathcal{R}}\left[E(\mathcal{S}, \mathcal{R})\right]$. To this end, we again apply the BDI (3) for function G and set Q being $\mathbb{E}_{\mathcal{R}}\left[E(\mathcal{S}, \mathcal{R})\right]$ and set of training samples $\mathcal{S}$, respectively. Note that, in this case, the bounded difference condition (2) equals the uniform stability (27), so $\rho = \beta$. Let $f_{\mathcal{S},\mathcal{R}}$ and $f_{\mathcal{S}',\mathcal{R}}$ be the two output models using the learning algorithm $\mathcal{A}$ with the set of random parameters $\mathcal{R}$ applied on two training sets $\mathcal{S}, \mathcal{S}'$, respectively. Assume $\mathcal{S}, \mathcal{S}'$ differ only in j-th element. For every $S, S', \mathcal{R}$, we have

$$\begin{aligned}
&\Bigg|\mathbb{E}_{\mathcal{R}}\left[E(\mathcal{S}, \mathcal{R})\right] - \mathbb{E}_{\mathcal{R}}\left[E(\mathcal{S}', \mathcal{R})\right]\Bigg| \\
&\quad = \Bigg|\mathbb{E}_{\mathcal{R}}\Bigg[\mathbb{E}_{\mathbf{z}\sim\mathcal{D}}\big[\ell(f_{\mathcal{S},\mathcal{R}}; \mathbf{z})\big] - \frac{1}{N}\sum_{i=1}^{N} \ell(f_{\mathcal{S},\mathcal{R}}; \mathbf{z}_i)\Bigg] \\
&\quad - \mathbb{E}_{\mathcal{R}}\Bigg[\mathbb{E}_{\mathbf{z}\sim\mathcal{D}}\big[\ell(f_{\mathcal{S}',\mathcal{R}}; \mathbf{z})\big] - \frac{1}{N}\sum_{i=1}^{N} \ell(f_{\mathcal{S}',\mathcal{R}}; \mathbf{z}_i)\Bigg]\Bigg| \\
&\quad = \Bigg|\mathbb{E}_{\mathbf{z}\sim\mathcal{D}}\Bigg[\mathbb{E}_{\mathcal{R}}\big[\ell(f_{\mathcal{S},\mathcal{R}}; \mathbf{z}) - \ell(f_{\mathcal{S}',\mathcal{R}}; \mathbf{z})\big]\Bigg] \\
&\quad - \frac{1}{N}\sum_{i=1}^{N} \mathbb{E}_{\mathcal{R}}\big[\ell(f_{\mathcal{S},\mathcal{R}}; \mathbf{z}_i) - \ell(f_{\mathcal{S}',\mathcal{R}}; \mathbf{z}_i)\big]\Bigg| \\
&\quad \underbrace{\leq \mathbb{E}_{\mathbf{z}\sim\mathcal{D}}\Bigg[\mathbb{E}_{\mathcal{R}}\big[\,|\ell(f_{\mathcal{S},\mathcal{R}}; \mathbf{z}) - \ell(f_{\mathcal{S}',\mathcal{R}}; \mathbf{z})|\,\big]\Bigg]}_{a} \\
&\quad \underbrace{+ \frac{1}{N}\sum_{i=1, i\neq j}^{N} \mathbb{E}_{\mathcal{R}}\big[\,|\ell(f_{\mathcal{S},\mathcal{R}}; \mathbf{z}_i) - \ell(f_{\mathcal{S}',\mathcal{R}}; \mathbf{z}_i)|\,\big]}_{b} \\
&\quad + \frac{1}{N}\mathbb{E}_{\mathcal{R}}\big[\,|\ell(f_{\mathcal{S},\mathcal{R}}; \mathbf{z}_i) - \ell(f_{\mathcal{S}',\mathcal{R}}; \mathbf{z}_i)|\,\big] \\
&\quad \leq 2\beta + \frac{L}{N},
\end{aligned} \tag{36}$$

where the terms (a) and (b) are upper bounded by β using the definition of uniform stability. Therefore, we have

$$\sup_{\mathcal{S},\mathcal{S}' \in \mathbb{R}^N} \Bigg|\mathbb{E}_{\mathcal{R}}\left[E(\mathcal{S}, \mathcal{R})\right] - \mathbb{E}_{\mathcal{R}}\left[E(\mathcal{S}', \mathcal{R})\right]\Bigg| \leq 2\beta + \frac{L}{N}. \tag{37}$$

Applying the BDI (3) results in the following inequality

$$\begin{aligned}
\mathbb{P}_{\mathcal{S}}\big[\mathbb{E}_{\mathcal{R}}\left[E(\mathcal{S}, \mathcal{R})\right] - \mathbb{E}_{\mathcal{S},\mathcal{R}}\left[E(\mathcal{S}, \mathcal{R})\right] \geq \epsilon\big] \\
\leq \exp(-2N\epsilon^2/(2N\beta + L)^2).
\end{aligned} \tag{38}$$

By setting the r.h.s. equal to ν, the following inequality holds with probability at least $1 - \nu$:

$$\mathbb{E}_{\mathcal{R}}\left[E(\mathcal{S}, \mathcal{R})\right] \leq \frac{(2N\beta + L)\sqrt{\log(1/\nu)}}{\sqrt{2N}} + \mathbb{E}_{\mathcal{S},\mathcal{R}}\left[E(\mathcal{S}, \mathcal{R})\right]. \tag{39}$$

Now, we provide the upper bound for $\mathbb{E}_{\mathcal{S},\mathcal{R}}\left[E(\mathcal{S}, \mathcal{R})\right]$. Denote by $\mathcal{T} = \{\mathbf{t}_i, i = 1, 2, \cdots, N\}$, a set of N training samples that are independent from $\mathcal{S}$ and are drawn from an unknown distribution $\mathcal{D}$. Denote $\mathcal{S}'$ the set obtained by replacing the

i-th sample in the set $\mathcal{S}$ with i-th sample from the set $\mathcal{T}$.

$$\begin{aligned}
&\mathbb{E}_{\mathcal{S},\mathcal{R}}\left[E(\mathcal{S},\mathcal{R})\right] \\
&\quad= \mathbb{E}_{\mathcal{S},\mathcal{R}}\left[\mathbb{E}_{\mathbf{z}\sim\mathcal{D}}\big[\ell(f_{\mathcal{S},\mathcal{R}};\mathbf{z})\big] - \frac{1}{N}\sum_{i=1}^{N}\ell(f_{\mathcal{S},\mathcal{R}};\mathbf{z}_i)\right] \\
&\quad= \mathbb{E}_{\mathcal{S},\mathcal{R}}\Big[\mathbb{E}_{\mathbf{z}\sim\mathcal{D}}\big[\ell(f_{\mathcal{S},\mathcal{R}};\mathbf{z})\big]\Big] \\
&\qquad\quad - \mathbb{E}_{\mathcal{S},\mathcal{T},\mathcal{R}}\left[\frac{1}{N}\sum_{i=1}^{N}\ell(f_{\mathcal{S},\mathcal{R}};\mathbf{t}_i)\right] \\
&\quad+ \mathbb{E}_{\mathcal{S},\mathcal{T},\mathcal{R}}\left[\frac{1}{N}\sum_{i=1}^{N}\ell(f_{\mathcal{S},\mathcal{R}};\mathbf{t}_i)\right] \\
&\qquad\quad - \mathbb{E}_{\mathcal{S},\mathcal{R}}\left[\frac{1}{N}\sum_{i=1}^{N}\ell(f_{\mathcal{S},\mathcal{R}};\mathbf{z}_i)\right] \\
&\quad= \mathbb{E}_{\mathcal{S},\mathcal{R}}\Big[\mathbb{E}_{\mathbf{z}\sim\mathcal{D}}\big[\ell(f_{\mathcal{S},\mathcal{R}};\mathbf{z})\big]\Big] \\
&\qquad\quad - \mathbb{E}_{\mathcal{S},\mathcal{R}}\Big[\mathbb{E}_{\mathbf{t}\sim\mathcal{D}}\big[\ell(f_{\mathcal{S},\mathcal{R}};\mathbf{t})\big]\Big] \\
&\qquad\quad + \mathbb{E}_{\mathcal{S},\mathcal{T},\mathcal{R}}\left[\frac{1}{N}\sum_{i=1}^{N}\ell(f_{\mathcal{S},\mathcal{R}};\mathbf{t}_i)\right] \\
&\qquad\quad - \mathbb{E}_{\mathcal{S},\mathcal{T},\mathcal{R}}\left[\frac{1}{N}\sum_{i=1}^{N}\ell(f_{\mathcal{S}',\mathcal{R}};\mathbf{t}_i)\right] \\
&\quad= \mathbb{E}_{\mathcal{S},\mathcal{T},\mathcal{R}}\left[\frac{1}{N}\sum_{i=1}^{N}\ell(f_{\mathcal{S},\mathcal{R}};\mathbf{t}_i)\right] \\
&\qquad\quad - \mathbb{E}_{\mathcal{S},\mathcal{T},\mathcal{R}}\left[\frac{1}{N}\sum_{i=1}^{N}\ell(f_{\mathcal{S}',\mathcal{R}};\mathbf{t}_i)\right] \\
&\quad= \mathbb{E}_{\mathcal{S},\mathcal{T},\mathcal{R}}\left[\frac{1}{N}\sum_{i=1}^{N}\big(\ell(f_{\mathcal{S},\mathcal{R}};\mathbf{t}_i) - \ell(f_{\mathcal{S}',\mathcal{R}};\mathbf{t}_i)\big)\right] \\
&\quad\le \sup_{\mathcal{S},\mathcal{S}',\mathbf{z}}\mathbb{E}_{\mathcal{R}}\big[\,|\ell(f_{\mathcal{S},\mathcal{R}};\mathbf{z}) - \ell(f_{\mathcal{S}',\mathcal{R}};\mathbf{z})|\,\big] \le \beta.
\end{aligned} \tag{40}$$

The last line is derived from the uniform stability definition (27) and amounts to changing $\mathbf{t}$ to $\mathbf{z}$. By combining (35), (39) and (40) , the following inequality holds with probability at least $1-2\nu$.

$$\begin{aligned}
&E(\mathcal{S},\mathcal{R}) \\
&\le \rho\sqrt{T\log(1/\nu)} + \beta\left(1+\sqrt{2N\log(1/\nu)}\right) + L\sqrt{\frac{\log(1/\nu)}{2N}}.
\end{aligned} \tag{41}$$

The results follows by setting $\delta = 2\nu$. ■

The stability parameter β and the BDI constant ρ depend on the properties of the learning algorithm $\mathcal{A}$ used for solving the minimisation problem (1). Considering SGD as the learning algorithm, we state the following theorem [10] which provides upper bounds for β and ρ.

Theorem 4. *Suppose that SGD update rule is executed for T iterations with an annealing learning rate λ_t to solve the optimisation problem* (1)*. If $\ell(f^\theta(\mathbf{x}),\mathbf{y})$ is convex, γ-Lipschitz and η-smooth with respect to its first argument for every $\mathbf{z}=(\mathbf{x},\mathbf{y})$, then SGD satisfies the property of being β-uniformly stable and holds the ρ-bounded difference condition* (2) *with respect to the loss function $\ell(f_{\mathcal{S},\mathcal{R}},\mathbf{z})$ and the set of random parameters $\mathcal{R}$. We have*

$$\begin{aligned}
\beta &\le \frac{2\gamma^2}{N}\sum_{t=1}^{T}\lambda_t \\
\rho &\le \frac{4\gamma^2}{T}\sum_{t=1}^{T}\lambda_t.
\end{aligned} \tag{42}$$

Proof. We first prove the first inequality which is similar to [10, Theorem 3.7]. We include the proof here for the sake of completeness. Then we show how to prove the second inequality. For the sake of simplicity of notation, we represent the output model $f^\theta_{\mathcal{S},\mathcal{R}}$ as $\theta_{\mathcal{S},\mathcal{R}}$ in this proof. We will omit $\mathcal{S},\mathcal{R}$ when it is clear from the context. Given a learning rate $\lambda \ge 0$ and a training set $\mathcal{S}$, SGD performs the gradient descent update rule, defined as $G(\theta) = \theta - \lambda\nabla_\theta\ell(\theta;\mathbf{z})$, T steps over all samples in $\mathcal{S}$. Here, sample $\mathbf{z}$ is randomly picked from $\mathcal{S}$. Assume the gradient update G is τ-expansive, *i.e.* $\sup_{u,v\in\mathcal{H}}\|\frac{G(u)-G(v)\|}{\|u-v\|} \le \tau$, and σ-bounded, *i.e.* $\sup_{\theta\in\mathcal{H}}\|\theta - G(\theta)\| \le \sigma$. Since $\ell(f^\theta(\mathbf{x}),\mathbf{y}) = \ell(<\theta,\mathbf{x}>,\mathbf{y})$ is convex, γ-Lipschitz and η-smooth with respect to its first argument for every $\mathbf{z}=(\mathbf{x},\mathbf{y})$, we have $\|\theta - G(\theta)\| \le \lambda\|\nabla_\theta\ell(\theta;\mathbf{z})\| = \lambda\|\nabla_\theta\ell(<\theta,\mathbf{x}>,\mathbf{y})\| \le \lambda\gamma$. Therefore the update rule is $\lambda\gamma$-bounded.

Let $\theta^1_{\mathcal{S}},\cdots\theta^T_{\mathcal{S}}$ and $\theta^1_{\mathcal{S}'},\cdots\theta^T_{\mathcal{S}'}$ be two sequences of output models resulting respectively from performing two sequences of the gradient updates $G(\theta^1_{\mathcal{S}}),\cdots G(\theta^T_{\mathcal{S}})$ and $G(\theta^1_{\mathcal{S}'}),\cdots G(\theta^T_{\mathcal{S}'})$ applied to two training sets $\mathcal{S},\mathcal{S}'$. Assume sets $\mathcal{S},\mathcal{S}'$ differ only in one element and the initialisation weights $\theta^0_{\mathcal{S}} = \theta^0_{\mathcal{S}'}$. Let $\Delta^t = \|\theta^t_{\mathcal{S}} - \theta^t_{\mathcal{S}'}\|$. The proof is based on the growth recursion concept [10, Lemma 2.4] which investigates how two distinct sequences of update rules applied to a deep neural model diverge when they start from the same initialisation point and the training set is perturbed at each step. For simplicity, we recall here the growth recursion result.

Growth recursion rule. *[10, Lemma 2.4] There exists the following relation recurrence between Δ^{t+1} and Δ^t:*

- *If $G(\theta^t_{\mathcal{S}})$ and $G(\theta^t_{\mathcal{S}'})$ are equal and τ-expansive, then $\Delta^{t+1} \le \tau\Delta^t$*
- *$G(\theta^t_{\mathcal{S}})$ and $G(\theta^t_{\mathcal{S}'})$ are σ-bounded and τ-expansive, then $\Delta^{t+1} \le \min(\tau,1)\Delta^t + 2\sigma_t$*

At each step t, there are two cases when two samples $\mathbf{z}$ and $\mathbf{z}'$ are picked by SGD from $\mathcal{S}$ and $\mathcal{S}'$ respectively: 1) $\mathbf{z}$ and $\mathbf{z}'$ are the same with probability $1-1/N$, which implies $G^t_{\mathcal{S}} = G^t_{\mathcal{S}'}$, 2) $\mathbf{z}$ and $\mathbf{z}'$ are different with probability $1/N$. Further, if the loss function is smooth and convex and the learning rate λ_t is small enough, it is proved that the gradient update rule $G^t_{\mathcal{S}}$ is 1-expansive [10]. Beside that, as mentioned before, $G^t_{\mathcal{S}}$ is $\lambda\gamma$-bounded. Applying the growth recursion rule results:

$$\Delta^{t+1} \le \left(1-\frac{1}{N}\right)\Delta^t + \frac{1}{N}\Delta^t + \frac{2\lambda\gamma}{N}. \tag{43}$$

Considering this inequality recursively through all steps, we obtain

$$\Delta^T \leq \frac{2\gamma}{N}\sum_{t=1}^{T}\lambda_t. \tag{44}$$

Using (43) and the fact that the loss function $\ell(f^\theta(\mathbf{x}),\mathbf{y}) = \ell(<\theta,\mathbf{x}>,\mathbf{y})$ is γ-Lipschitz with respect to its first argument, the following inequality is obtained for any $\mathbf{z}$, $\mathcal{S}$, $\mathcal{S}'$:

$$\begin{aligned}
&\mathbb{E}_{\mathcal{R}}\left[\left|\ell(\theta^T_{\mathcal{S},\mathcal{R}};\mathbf{z}) - \ell(\theta^T_{\mathcal{S}',\mathcal{R}};\mathbf{z})\right|\right]\\
&\quad\leq \mathbb{E}_{\mathcal{R}}\left[\left|\ell(<\theta^T_{\mathcal{S},\mathcal{R}},\mathbf{x}>,\mathbf{y})) - \ell(<\theta^T_{\mathcal{S}',\mathcal{R}},\mathbf{x}>,\mathbf{y}))\right|\right]\\
&\quad\leq \gamma\mathbb{E}_{\mathcal{R}}\left[\left\|<\theta^T_{\mathcal{S},\mathcal{R}},\mathbf{x}> - <\theta^T_{\mathcal{S}',\mathcal{R}},\mathbf{x}>\right\|\right]\\
&\quad\leq \gamma\mathbb{E}_{\mathcal{R}}\left[\left\|\theta^T_{\mathcal{S},\mathcal{R}} - \theta^T_{\mathcal{S}',\mathcal{R}}\right\|\right] = \gamma\mathbb{E}_{\mathcal{R}}\left[\Delta^T\right] \leq \frac{2\gamma^2}{N}\sum_{t=1}^{T}\lambda_t,
\end{aligned} \tag{45}$$

where the expectation is taken over randomness of the SGD algorithm, which appears in the random set $\mathcal{R}$. Without loss of generality, we assume $\|\mathbf{x}\| \leq 1$ in the last inequality. The inequality (45) implies the uniform stability (27), which renders the desired inequality.

Now, we derive the second inequality. The proof follows the same reasoning as that used for deriving the first inequality, except that the sequences of the update rule relate to $\mathcal{R},\mathcal{R}'$. Let $\theta^1_{\mathcal{R}},\cdots\theta^T_{\mathcal{R}}$ and $\theta^1_{\mathcal{R}'},\cdots\theta^T_{\mathcal{R}'}$ be two sequences of output models resulting respectively from performing two sequences of the gradient updates $G(\theta^1_{\mathcal{R}}),\cdots G(\theta^T_{\mathcal{R}})$ and $G(\theta^1_{\mathcal{R}'}),\cdots G(\theta^T_{\mathcal{R}'})$ applied to the training sets $\mathcal{S}$ with two different random sets $\mathcal{R},\mathcal{R}'$. Assume sets $\mathcal{R},\mathcal{R}'$ differ only at two element and the initialisation weights $\theta^0_{\mathcal{S}} = \theta^0_{\mathcal{S}'}$. Let $\Delta^t = \|\theta^t_R - \theta^t_{R'}\|$. At each step t, there are two cases when two samples $\mathbf{z}$ and $\mathbf{z}'$ are picked by SGD by the permutation order in $\mathcal{R}$ and $\mathcal{R}'$ respectively: 1) $\mathbf{z}$ and $\mathbf{z}'$ are the same with probability $1-2/N$, which implies $G^t_R = G^t_{R'}$, 2) $\mathbf{z}$ and $\mathbf{z}'$ are different with probability $2/N$. Following the same chain of equations (43)-(45) results in:

$$\begin{aligned}
&\left|\ell(\theta^T_{\mathcal{S},\mathcal{R}};\mathbf{z}) - \ell(\theta^T_{\mathcal{S},\mathcal{R}'};\mathbf{z})\right|\\
&\quad\leq \left|\ell(<\theta^T_{\mathcal{S},\mathcal{R}},\mathbf{x}>,\mathbf{y})) - \ell(<\theta^T_{\mathcal{S},\mathcal{R}'},\mathbf{x}>,\mathbf{y}))\right|\\
&\quad\leq \gamma\left\|<\theta^T_{\mathcal{S},\mathcal{R}},\mathbf{x}> - <\theta^T_{\mathcal{S},\mathcal{R}'},\mathbf{x}>\right\|\\
&\quad\leq \gamma\left\|\theta^T_{\mathcal{S},\mathcal{R}} - \theta^T_{\mathcal{S},\mathcal{R}'}\right\| = \gamma\Delta^T \leq \frac{4\gamma^2}{T}\sum_{t=1}^{T}\lambda_t,
\end{aligned} \tag{46}$$

The inequality (46) implies the desired inequality. ■

In the previous section, we have proved that the GJM, JS and KL loss functions satisfy the Lipschitz continuity and smoothness properties. So, the upper bounds (42) are valid for SGD when these loss functions are used[2]. We can now get the following theorem by combining Theorem 3 and Theorem 4.

Theorem 5. *Consider a loss function ℓ such that $0 \leq \ell(f(\cdot;\mathbf{z}) \leq L$ for any point $\mathbf{z}$. Suppose that SGD update rule is executed for T iterations with an annealing learning rate λ_t to solve the optimisation problem* (1)*. Then, we have the following generalisation error bound with probability at least $1-\delta$:*

$$\begin{aligned}
&R_{\text{true}}(f_{\mathcal{S},\mathcal{R}}) - R_{\text{emp}}(f_{\mathcal{S},\mathcal{R}})\\
&\quad\leq 2\gamma^2\sum_{t=1}^{T}\lambda_t\left(2\sqrt{\frac{\log(2/\delta)}{T}} + \sqrt{\frac{2\log(2/\delta)}{N}} + \frac{1}{N}\right)\\
&\quad\quad + L\sqrt{\frac{\log(2/\delta)}{2N}}.
\end{aligned} \tag{47}$$

Remark. Theorem 5 implies that the generalisation error decreases inversely with the size of the training set. The first term in (47) improves the bound due to the Lipschitz property of the loss function and vanishes as γ decreases. The second term in (47) depends on the maximum value that the loss function can assume. Both, the first and second terms can be controlled by the type of loss function used.

We now assess the impact of the above-mentioned loss functions on controlling the uniform stability and the generalisation error bound. Specifically, we are interested in proving that the generalisation error bound of a model, being trained by the GJM loss function, gets the tighter upper bound in comparison with the models trained by the JS and KL loss functions. The following corollary compares the generalisation error bounds for the learning algorithm which uses different loss functions.

Corollary 2. *Consider a DNN is trained using the GJM loss, KL loss and CE loss functions, separately over the same training set $\mathcal{S}$ and settings. Denote $f^{GJM}_{\mathcal{S},\mathcal{R}}, f^{JS}_{\mathcal{S},\mathcal{R}}, f^{KL}_{\mathcal{S},\mathcal{R}}$ as the corresponding output models. We have the following inequalities:*

$$E(f^{GJM}_{\mathcal{S},\mathcal{R}}) \leq E(f^{JS}_{\mathcal{S},\mathcal{R}}) \leq E_{KL}(f^{KL}_{\mathcal{S},\mathcal{R}}), \tag{48}$$

where $E(f_{\mathcal{S},\mathcal{R}}) = R_{\text{true}}(f_{\mathcal{S},\mathcal{R}}) - R_{\text{emp}}(f_{\mathcal{S},\mathcal{R}})$.

Proof. These inequalities immediately follow from Corollary 1. Regarding the last term, note the JS and KL loss functions are upper bounded by $\log_2 K$ and ∞, respectively [45] and the GJM is upper bounded by the JS loss function according to Theorem 2. ■

Corollary 2 provides the tighter bound on the generalisation error of SGD when GMD is used as the loss function. In other words, the generalisation error of a model trained with the proposed loss function is upper bounded by the generalisation error obtained by the other loss functions.

VI. Experimental Evaluation

In this section, the theoretical findings obtained by our analysis in the previous sections are experimentally validated across a variety of tasks. We focus on applications where there exists semantic correlation between classes, *i.e.* facial age estimation, head pose estimation and image aesthetics assessment. None of these results are intended to represent the state-of-the-art for any particular task — our goal is to demonstrate how the loss function affects the generalisation performance of a learnt DNN model trained by SGD. We will

[2] Modern results, obtained under some strong assumptions such as strong convexity of the loss function [10], [43], [44], bound the generalisation gap tighter. One may expect that designing a strongly convex loss function would improve the generalisation ability of a trained model by SGD.

show that across a variety of tasks, replacing the loss function with our proposed loss functions can provide a significant improvement in the generalisation performance of the trained models.

A. Settings

Our algorithms are implemented with MatConvNet framework [46]. In all experiments, the VGG model [47] is used as the backbone of the system. We use the VGG model pre-trained on face recognition datasets [48] for the age prediction and head pose estimation tasks. For the image aesthetic assessment, we train the VGG model from scratch. We replace the last class fully-connected (FC) layer in the VGG model with a K-neurons FC layer, where K is the number of classes in the target task. We set the number of age, pose and aesthetic score classes to 101, 61 and 10, respectively, The FC layer's weights are randomly initialised. For the experiments on all the tasks, we use the following settings: 1. The batch size, parameter α, initial learning rate, weight decay and momentum are set to 64, 0.5, 0.001, 0.0005 and 0.9 respectively. 2. The learning rate is decreased exponentially to reach 10^{-5} over 30 epochs. 3. We do online augmentation by performing random cropping and flipping of images during the training phase.

To measure the generalisation performance, we follow the cross-dataset evaluation setting introduced in [12]. We keep aside several test sets $\mathcal{T} \in (\mathcal{X} \times \mathcal{Y})^M$, so that the model $f_{\mathcal{S},\mathcal{R}}$, which has been picked by the learning algorithm $\mathcal{A}$ using the training set $\mathcal{S}$, is statistically independent of $\mathcal{T}$s. For the age estimation and head pose estimation tasks, we also guarantee no overlap between the sets of subjects in the training and test sets. Under these constraints, we enforce the trained models to be blind to the characteristics of the images in the test datasets. Therefore, the generalisation performance of the trained models in unseen and uncontrolled scenarios can be more reliably evaluated.

B. Baselines

We consider several baseline to validate the performance of our models. The performance is first compared with the standard classification based framework [49], where the labels are one-hot encoded and the cross entropy (CE) is used as the loss function for training the network. It should be noted that the CE loss function is a special case of the KL loss, because $\ell_{KL}(\hat{\mathbf{y}}, \mathbf{y}) = \ell_{CE}(\hat{\mathbf{y}}, \mathbf{y}) + H(\mathbf{y})$, where $H(\mathbf{y})$ is the negative entropy of $\mathbf{y}$, which is constant. Further, when $\sigma \to 0$, the KL loss will approach the CE loss. Within the LDL framework, we also evaluate the performance of the model trained via the KL divergence [14] and χ^2-statistic [50] employed as the loss function. For a fair comparison, we trained the models on a training dataset with the same network architecture and implementation settings, including the pre-processing steps and the data augmentation techniques. Finally, we investigate the merit of the JS divergence and our proposed loss function ($\alpha = 0.5$) in the context of deep LDL based systems, compared with the above-mentioned algorithms.

C. Facial Age Estimation

A typical facial age estimation algorithm maps a face image into its corresponding chronological age label, a real number between 0 to 100. Its aim is to learn discriminative features so that the feature space can be divided into homogeneous partitions, one for each age class. However, due to the fact that the face images of the adjacent ages may have quite similar facial appearance, the facial feature spaces across ages are heavily overlapping. In order to deal with such semantic correlation among classes in the age estimation problem, some recent work in the literature [14], [51] model the age estimation problem as the LDL problem, by which each age label is encoded as a label distribution $\mathbf{y} = [y_0, y_1, \cdots, y_{100}]$, where each element of y_i is assumed to be a real number in the range $[0, 1]$ and all elements are constrained to $\sum_{k=1}^{K} y_k = 1$. The highest value is at the true chronological age and the probabilities gradually decrease on both sides of the chronological age. As seen in Fig. 1, all ages in the range 20 to 40 can describe the 30-years old face image to varying extent. To transfer the age estimation problem into the LDL framework, we follow the same strategy used in [14], [51] and generate a Gaussian distribution, centred at the age label with a standard deviation $\sigma = 2$, for each face image.

Training Set: We combine the two existing ageing datasets, namely AgeDB [52] and UTKFace [53], for building the training set $\mathcal{S}$. The AgeDB [52] and the UTKFace [53] datasets contain $16,488$ and $21,374$ images, respectively. Due to limited number of images in very young and old ages, we also crawled a set of $23,876$ facial images from the Internet in the range of $0-20$ years and $70-100$ years. We added this set of images to the training set. In contrast to the above-mentioned datasets, our training set has enough images with the age labels, ranging from 0 to 100.

Test Sets: MORPH [55], FG-NET [56], FACES [57] and SC-FACE [58] datasets are used as the test sets. MORPH dataset contains $55,134$ images from $13,617$ subjects of different races in the age range from 16 to 72 years old. It provides a suitable dataset for analysing the generalisation performance because most of images in the dataset are African people, while this ethnic group is under represented in our training dataset. The FG-NET dataset contains $1,002$ images of 82 subjects in the age range from 0 to 69 years old. It is challenging due to its large variations in pose, expression and lighting conditions. The FACES dataset has $2,052$ images of 171 subjects with six expressions (neutrality, happiness, anger, fear, disgust, and sadness) in the age range from 19 to 80 years old. SC-FACE dataset contains $4,160$ images of 130 subjects in the age range from 21 to 75 years old. We separate this dataset into two separate datasets, namely SC-FACE-ROT and SC-FACE-SUR datasets, which contain $1,170$ and $2,990$ images, respectively. Taken by a digital high-quality camera, each subject in the SC-FACE-ROT dataset has one high-resolution frontal image and 9 images with different head poses ranging from $-90°$ to $+90°$ in equal steps of $22.5°$. Each subject in the SC-FACE-SUR dataset has 17 images with different qualities.

We apply two pre-processing steps to all images in the training and test sets. First, face bounding box and 5 facial

TABLE I
AGE ESTIMATION EVALUATION (MAE & CS) ON THE TEST DATASETS.

	FG-NET		MORPH		FACES		SC-FACE-ROT		SC-FACE-SUR		Average	
Method	MAE	CS (%)	MAE	CS (%)	MAE	CS (%)	MAE	CS (%)	MAE	CS (%)	MAE	CS (%)
Human Workers [54]	4.70	69.5	6.30	51.0	NA	NA	NA	NA	NA	NA	5.50	60.25
CE [49]	3.57	78.94	6.54	53.38	6.59	50.83	6.45	49.32	6.19	65.05	5.86	59.50
KL [14]	3.24	81.54	6.01	57.36	6.11	55.60	5.90	54.79	6.52	60.64	5.55	61.98
χ^2	3.29	80.44	5.98	56.10	6.05	55.77	5.61	58.55	5.75	66.89	5.33	63.55
JS [ours]	3.33	79.74	5.64	58.17	6.49	53.27	5.37	62.14	**5.11**	**68.73**	5.18	64.41
GJM [ours]	**3.21**	**81.59**	**5.63**	**59.13**	**5.90**	**57.55**	**5.32**	**62.14**	5.37	67.96	**5.08**	**65.67**

landmarks of each face (*i.e.* the left and right centre of the eyes, the nose tip, the left and right corner of the mouth) are extracted by applying the MTCNN face detector [59] on each image of the training and test sets. Second, the alignment method, proposed in [60], is used to align a face in the centre of the input image to the DNN. Finally, the aligned image is squeezed to 256×256 pixels.

Evaluation: At the evaluation step, we use the central cropped image as the input to the network. To demonstrate the generalisation performance of age estimation systems, we calculate the mean absolute error (MAE) defined as $\sum_{k=1}^{M} \frac{|\hat{l}_k - l_k|}{M}$, where M is the total number of test images and $\hat{l}_k$ is the predicted age of the l-th image obtained by taking the age corresponding to the maximum value of the output distribution of the DNN. We also report the cumulative score (CS) [61] which is defined as $\frac{M_I}{M} \times 100\%$, where M_I is the number of images such that $|\hat{y}_k - y_k| < I$. In this paper, we set I as 5.

In Table I, we evaluate the performance of several age estimation systems in terms of MAE and CS measures. We can see that the age estimation accuracy of the LDL based methods is higher than those of the classification based method. This indicates that utilising label distribution is helpful to improve the age estimation performance. This is reasonable because the classification based model does not consider the effect of the correlation during the training process. Further, it can be inferred from Tables I that adopting the GJM loss function for the age estimation leads to higher prediction accuracy compared to the prediction accuracy achieved by the other loss functions, including KL divergence, χ^2-statistic, and JS divergence. Other things being equal, the GJM loss leads to a generalisation performance which is practically distinguishable from the JS loss function and much better than the KL loss function.

D. Head Pose Estimation

Head pose estimation is useful in a wide variety of applications, such as behaviour analysis, gaze estimation, fatigue driving detection, and face recognition [62]. A typical head pose estimation algorithm predicts pitch and yaw angles which are real numbers between $-90°$ to $+90°$. Due to the scarcity of datasets that are annotated with both, yaw and pitch angles, we focus on the yaw angle prediction in this paper. Similar to the age estimation problem, the face images of the close poses are quite similar in facial appearance. Thus, the facial feature spaces across poses are heavily overlapping. By virtue of considering a label distribution for each pose label, the correlation among neighbouring poses are effectively taken into consideration during the training process [14]. To transfer the head pose estimation problem into the LDL framework, we follow the same strategy adopted in the age estimation problem. We generate a Gaussian label distribution, centred at the ground-truth of head pose label (yaw angle) with a standard deviation $\sigma = 3$, for each face image. Fig. 1 shows a label distribution with the yaw angle = $60°$.

Training Set: The AFLW dataset [63] is used for training. It contains about $24k$ in-the-wild face images. We select $22,490$ faces to ensure the yaw angles fall within the range from $-90°$ to $+90°$. The ground-truth head pose (yaw) angles are given as real numbers from $-90°$ to $+90°$ in steps of $3°$. So, we have 61 yaw categories in this dataset.

Test Sets: We evaluate the generalisation performance of the proposed head pose predictors on four public datasets: Pointing'04 [64], NCKU [65], SC-FACE [58], MULTI-PIE [66] and BIWI [67]. The Pointing'04 dataset contains $2,790$ images of 15 subjects. The images display variations in expressions, skin colours, and occlusions (*e.g.*wearing glasses). The yaw angles range between $-90°$ and $+90°$ with increments of $15°$. The NCKU dataset contains $6,660$ images of 90 subjects (78 males and 12 females). Each subject has 74 images, where 37 images were taken every $5°$ from $-90°$ to $+90°$. The SC-FACE dataset contains $1,170$ images of 130 subjects with different head poses ranging from $-90°$ to $+90°$ in equal steps of $22.5°$. The MULTI-PIE dataset is a collection of faces of 337 subjects. The yaw angles range between $-90°$ and $+90°$ with increments of $15°$. Images in this dataset exhibit variation through differences in illumination and facial expressions. For our evaluation, we randomly selected 100 subjects from the original MULTI-PIE dataset. BIWI Kinect dataset contains over $15K$ images of 20 subjects. The head pose range covers between $-75°$ and $+75°$.

For all images in the training and test sets, the faces are first detected in the input images by using the MTCNN face detector [59]. They are then cropped from the original images and normalised to images of 256×256 pixels.

Evaluation: Given an input image, we determine the bin of the output pose distribution with the maximum value as the predicted yaw angle. The MAE and CS scores are used for measuring the performance of the trained head pose estimators.

TABLE II
HEAD POSE ESTIMATION EVALUATION (MAE & CS) ON THE TEST DATASETS.

	Pointing'04		NCKU		SC-FACE		MULTI-PIE		BIWI		Average	
Method	MAE	CS (%)	MAE	CS (%)	MAE	CS (%)	MAE	CS (%)	MAE	CS (%)	MAE	CS (%)
CE [49]	15.10	55.95	12.31	73.24	10.59	75.04	12.25	78.25	15.45	54.44	13.14	67.38
χ^2	12.77	58.80	12.01	71.33	10.67	71.70	10.56	75.10	15.57	53.76	12.31	66.13
KL [14]	13.00	59.52	12.68	72.07	7.53	85.21	8.39	86.90	15.41	55.01	11.40	71.74
JS [ours]	12.81	60.71	**11.34**	**78.85**	9.57	80.08	9.69	79.70	15.10	55.92	11.70	71.05
GJM [ours]	**12.40**	**61.42**	11.52	78.60	**7.04**	**89.05**	**7.86**	**89.85**	**14.95**	**56.12**	**10.75**	**75.00**

We set I as 15. Table II reports the results achieved by the different methods on the test datasets. We observe that using the GJM loss function for training DNN achieves the best performance: it has much lower MAE and higher accuracy than the other methods. As can be seen, the performance on the BIWI dataset is nearly the same for all competing methods. It is worth mentioning that the subjects in this dataset were recorded while turning their heads, sitting in front of the sensor, at roughly one meter of distance. Consequently, the head pose angles in this dataset are not very precise. Nevertheless, the performance of the model trained by our proposed method on this dataset is slightly better than the others.

E. Image Aesthetics Assessment

The image aesthetics assessment problem involves rating images on the basis of the subjective impression felt by several viewers when looking at them [68]–[71]. Each user gives a score to each viewed images captured by different devices, reflecting their opinion as to whether the photo has been acquired by an expert photographer, the scene context etc. To transform the image aesthetics assessment into the LDL framework, we adopt, as the label distribution, the normalised histogram of human opinion scores of each image. The label distributions in this application are mixture distributions, as shown in Fig. 1. Our approach for the image aesthetics assessment is to predict the label distribution. We then compute the expected (mean) value of the label distribution over the aesthetic bins of a given image as the aesthetic score.

Training and Test Sets: We train the VGG model on a large publicly available aesthetic assessment dataset, called aesthetic visual analysis (AVA) dataset [72]. The AVA dataset contains about $255,000$ images, collected from an amateur photography contest site[3]. The aesthetic quality of each image was rated by about 200 human annotators. The aesthetic score ranges from 1 to 10 according to the viewers' aesthetic judgements, with 10 indicating the highest quality. Following the cross-dataset setting [12], we train the VGG model on all the images in the AVA dataset and then evaluate the performance on two other datasets, namely AADB [73] and FLICKER-AES [74]. The AADB dataset contains $10,000$ photographic images of real scenes collected from Flickr. Each image was annotated with an aesthetic score for eleven attributes, averaged by five annotators. Aesthetic scores range from 0 to 1 with 1 denoting the highest quality. The FLICKER-AES dataset contains $40,000$ images. Aesthetic scores range from 1 to 5, representing the lowest to the highest aesthetic levels. Each image was rated by about five annotators and the corresponding aesthetic score is set to be the mean of these scores. These datasets differ from AVA which contains a significant number of professional images that have been highly manipulated, covered with advertising text, etc.

Evaluation: We evaluate the performance of image aesthetics grading in terms of ρ value [73] and accuracy score [75]. The ρ value is defined as $\rho = 1 - \frac{\sum_{k=1}^{M} l_k - \hat{l}_k}{M^3 - M}$, where M is the total number of test images. l_k and $\hat{l}_k$ are the ground-truth and predicted aesthetic scores of the l-th image obtained by computing the expected values of the predicted and ground-truth label distributions over the 10 aesthetic bins. The accuracy (Acc.) is measured by following the approach proposed in [75]. We compute the expected ground-truth and predicted aesthetic scores according to the ground-truth and the predicted label distributions. The accuracy is then assessed by means of a binary categorisation of the aesthetic scores. Images are categorised as high quality, if their expected score is greater than the cut-off score of 5. The Acc. measure is then defined as $\frac{M_I}{M} \times 100\%$, where M_I is the number of images such that $\hat{l}_k = l_k$.

The Acc. and ρ values of used in our evaluation on the AVA dataset are reported in Table III. As shown in Table III, our proposed loss function outperforms others in terms of both, the Acc. measure and the ρ value. It should be noted that the performance of image aesthetic assessment significantly drops under cross-dataset evaluation. This confirms that the models trained on the AVA dataset have a very limited generalisation capability. We conjecture that there are two reasons doe this behaviour. First, the AVA and test datasets are annotated by different groups of raters who might have different aesthetics tastes. Second, the AVA datasets contain photos with different distributions of visual characteristics than those of the AADB and FLICKER-AES datasets. Many images in the AVA datasets are professionally photographed, while the images in the AADB dataset contain many daily life photos from casual users. This observation establishes the need for a further exploration into the mechanisms for learning aesthetic rates, that would enable adaptation to the tastes of a variety of user groups and the diverse content of photo collections. As can be seen in Table III, our system exhibits a better generalisation capability compared with the existing ones.

[3] http://www.dpchallenge.com/

TABLE III
Image Aesthetics Assessment (ρ Value & Acc. Measure) on the Test Datasets.

	AADB		FLICKER-AES		Average	
Method	ρ	Acc.(%)	ρ	Acc.(%)	ρ	Acc.(%)
CE [49]	0.319	58.45	0.228	49.12	0.273	53.78
χ^2	0.320	58.61	0.238	49.45	0.279	54.03
KL [14]	0.322	58.76	0.242	49.44	0.282	54.10
JS [ours]	0.324	58.99	0.252	49.78	0.288	54.38
GJM [ours]	**0.327**	**60.32**	**0.261**	**50.10**	**0.294**	**55.21**

F. Discussions

α is the hyper-parameter in the proposed GJM loss function which affects the performance of the trained model. In the above experiments, we have empirically set $\alpha = 0.5$. In order to study the impact of α, we evaluate the generalisation performance with different α values, changing from 0 to 1. Here, we take the age estimation and head pose estimation problems as examples. Table IV shows the MAE performance of the age and head pose estimators on the FG-NET and Pointing'04 datasets with respect to different α. We can see that a proper α is important for low MAE. But generally speaking, an α value that is close to 0.5 is a good choice. Finally, it should be noted that using one GPU TITAN X 12 GB, the training time of one epoch of all methods discussed in this paper, over a training dataset, including 50K images of size 224×224 pixels, all takes 639 seconds on average. At the inference time, the output is predicted in 0.001 seconds.

TABLE IV
The Influence of Values of Parameter α

α	0.2	0.3	0.4	0.5	0.6	0.7	0.8
FG-NET	3.7	3.3	3.2	**3.2**	3.5	4	4.8
Pointing'04	13.9	13.0	12.8	**12.8**	12.9	13.1	15.25

VII. Conclusion

Our main goal in this paper was to theoretically study the effect of loss function on the generalisation properties of a DNN model trained by the stochastic gradient descent (SGD). We proved that SGD towards a DNN model, trained with an appropriate loss function, exhibits a stronger uniform stability, and this results in a tighter bound on the generalisation error. The practical benefit of the stronger stability is better generalisation of the resulting machine learning algorithms. Accordingly, we proposed a novel loss function for which we proved the tighter generalisation bound using the notion of uniform stability. We experimentally validated our theoretical findings by comparing the generalisation performance of different models, trained with diverse loss functions, including the proposed loss function, on a variety of class-correlated learning tasks formulated using the well-known label distribution learning (LDL) framework. The main outcome of our theoretical and experimental analyses is that the proposed loss function appears to be the criterion of choice for deep semantics-preserving learning tasks.

Acknowledgment

This work was supported in part by the EPSRC Programme Grant (FACER2VM) EP/N007743/1 and the EPSRC/dstl/MURI project EP/R018456/1. This project made use of time on Tier 2 HPC facility JADE2, funded by EPSRC (EP/T022205/1).

References

[1] A. Akbari, M. Awais, M. Bashar, and J. Kittler, "How does loss function affect generalization performance of deep learning? application to human age estimation," in *International Conference on Machine Learning (ICML)*, 2021, pp. 1–9.

[2] S. S. Khalid, M. Awais, Z. H. Feng, C. H. Chan, A. Farooq, A. Akbari, and J. Kittler, "Resolution invariant face recognition using a distillation approach," *IEEE Transactions on Biometrics, Behavior, and Identity Science*, vol. 2, no. 4, pp. 410–420, 2020.

[3] A. Akbari, M. Awais, Z. Feng, A. Farooq, and J. Kittler, "A flatter loss for bias mitigation in cross-dataset facial age estimation," in *International Conference on Pattern Recognition (ICPR)*, 2021.

[4] M. Bashar, A. Akbari, K. Cumanan, H. Q. Ngo, A. G. Burr, P. Xiao, and M. Debbah, "Deep learning-aided finite-capacity fronthaul cell-free massive mimo with zero forcing," in *IEEE International Conference on Communications (ICC)*, 2020, pp. 1–6.

[5] M. Bashar, A. Akbari, K. Cumanan, H. Q. Ngo, A. G. Burr, P. Xiao, M. Debbah, and J. Kittler, "Exploiting deep learning in limited-fronthaul cell-free massive mimo uplink," *IEEE Journal on Selected Areas in Communications (JSAC)*, vol. 38, no. 8, pp. 1678–1697, 2020.

[6] A. Akbari, M. Awais, and J. Kittler, "Sensitivity of age estimation systems to demographic factors and image quality: Achievements and challenges," in *IEEE International Joint Conference on Biometrics (IJCB)*, 2020, pp. 1–6.

[7] S. S. Khalid, M. Awais, C.-H. Chan, Z. Feng, A. Farooq, A. Akbari, and J. Kittler, "Npt-loss: A metric loss with implicit mining for face recognition," 2021.

[8] A. Elisseeff, T. Evgeniou, and M. Pontil, "Stability of randomized learning algorithms," *The Journal of Machine Learning Research*, vol. 6, pp. 55–79, Jan 2005.

[9] O. Bousquet and A. Elisseeff, "Stability and generalization," *J. Mach. Learn. Res.*, vol. 2, pp. 499–526, Mar. 2002.

[10] M. Hardt, B. Recht, and Y. Singer, "Train faster, generalize better: Stability of stochastic gradient descent," in *Proceedings of The 33rd International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 48, New York, New York, USA, Jun 2016, pp. 1225–1234.

[11] X. Geng, C. Yin, and Z. Zhou, "Facial age estimation by learning from label distributions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 10, pp. 2401–2412, Oct 2013.

[12] A. Akbari, M. Awais, Z. Feng, A. Farooq, and J. Kittler, "Distribution cognisant loss for cross-database facial age estimation with sensitivity analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2020.

[13] A. Akbari, M. Awais, S. Fatemifar, S. S. Khalid, and J. Kittler, "A novel ground metric for optimal transportbased chronological age estimation," *IEEE Transactions on Cybernetics*, pp. 1–1, 2021.

[14] B. Gao, C. Xing, C. Xie, J. Wu, and X. Geng, "Deep label distribution learning with label ambiguity," *IEEE Transactions on Image Processing*, vol. 26, no. 6, pp. 2825–2838, June 2017.

[15] S. Fatemifar, M. Awais, A. Akbari, and J. Kittler, "A stacking ensemble for anomaly based client-specific face spoofing detection," in *2020 IEEE International Conference on Image Processing (ICIP)*, 2020, pp. 1371–1375.

[16] A. Akbari, M. Trocan, S. Sanei, and B. Granado, "Joint sparse learning with nonlocal and local image priors for image error concealment," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 8, pp. 2559–2574, 2020.

[17] A. Akbari, M. Trocan, and B. Granado, "Image error concealment based on joint sparse representation and non-local similarity," in *IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, 2017, pp. 6–10.

[18] ——, "Joint-domain dictionary learning-based error concealment using common space mapping," in *International Conference on Digital Signal Processing (DSP)*, 2017, pp. 1–5.

[19] ——, "Image error concealment using sparse representations over a trained dictionary," in *Picture Coding Symposium (PCS)*, 2016, pp. 1–5.

[20] S. Shalev-Shwartz, O. Shamir, N. Srebro, and K. Sridharan, “Learnability, stability and uniform convergence,” *J. Mach. Learn. Res.*, vol. 11, pp. 2635–2670, Dec. 2010.

[21] L. Devroye and T. Wagner, “Distribution-free performance bounds for potential function rules,” *IEEE Transactions on Information Theory*, vol. 25, no. 5, pp. 601–604, 1979.

[22] S. Lin, “Generalization and expressivity for deep nets,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 5, pp. 1392–1406, 2019.

[23] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: A simple way to prevent neural networks from overfitting,” *Journal of Machine Learning Research*, vol. 15, no. 56, pp. 1929–1958, 2014.

[24] S. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, “Deepfool: A simple and accurate method to fool deep neural networks,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 2574–2582.

[25] C. Li, M. Georgiopoulos, and G. C. Anagnostopoulos, “Multitask classification hypothesis space with improved generalization bounds,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 7, pp. 1468–1479, 2015.

[26] C. Zhang, D. Tao, T. Hu, and B. Liu, “Generalization bounds of multitask learning from perspective of vector-valued function learning,” *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–14, 2020.

[27] S. Seong, Y. Lee, Y. Kee, D. Han, and J. Kim, “Towards flatter loss surface via nonmonotonic learning rate scheduling,” in *The Conference on Uncertainty in Artificial Intelligence (UAI)*, 2018.

[28] Y. Wang, Z. Bian, J. Hou, and L. Chau, “Convolutional neural networks with dynamic regularization,” *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–6, 2020.

[29] J. Yang, X. Zeng, S. Zhong, and S. Wu, “Effective neural network ensemble approach for improving generalization performance,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 24, no. 6, pp. 878–887, 2013.

[30] F. He, T. Liu, and D. Tao, “Why resnet works? residuals generalize,” *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–14, 2020.

[31] A. Maurer and M. Pontil, “Structured sparsity and generalization,” *Journal of Machine Learning Research*, vol. 13, no. 23, pp. 671–690, 2012.

[32] A. Akbari, M. Trocan, and B. Granado, “Sparse recovery-based error concealment,” *IEEE Transactions on Multimedia*, vol. 19, no. 6, pp. 1339–1350, 2017.

[33] D. Jakubovitz, R. Giryes, and M. R. D. Rodrigues, “Generalization error in deep learning,” *CoRR*, vol. abs/1808.01174, 2018. [Online]. Available: http://arxiv.org/abs/1808.01174

[34] X. Wu, J. Zhang, and F. Wang, “Stability-based generalization analysis of distributed learning algorithms for big data,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 3, pp. 801–812, 2020.

[35] V. Cherkassky, Xuhui Shao, F. M. Mulier, and V. N. Vapnik, “Model complexity control for regression using vc generalization bounds,” *IEEE Transactions on Neural Networks*, vol. 10, no. 5, pp. 1075–1089, 1999.

[36] H. Xu and S. Mannor, “Robustness and generalization,” *Machine Learning*, vol. 86, no. 3, pp. 391–423, Mar 2012.

[37] B. Neyshabur, S. Bhojanapalli, and N. Srebro, “A PAC-bayesian approach to spectrally-normalized margin bounds for neural networks,” in *International Conference on Learning Representations*, 2018. [Online]. Available: https://openreview.net/forum?id=Skz_WfbCZ

[38] C. McDiarmid, *On the method of bounded differences*, ser. London Mathematical Society Lecture Note Series. Cambridge: Cambridge University Press, 1989.

[39] S.-H. Cha, “Comprehensive survey on distance/similarity measures between probability density functions,” *International Journal of Mathematical Models and Methods in Applied Sciences*, vol. 1, no. 1, pp. 300–307, Nov 2007.

[40] A. L. Gibbs and F. E. Su, “On choosing and bounding probability metrics,” *International Statistical Review*, vol. 70, no. 3, pp. 419–435, 2002.

[41] L. L. Cam, *Asymptotic Methods in Statistical Decision Theory*, ser. Springer Series in Statistics. New York: Springer-Verlag, 1986.

[42] F. Topsoe, “Some inequalities for information divergence and related measures of discrimination,” *IEEE Transactions on Information Theory*, vol. 46, no. 4, pp. 1602–1609, 2000.

[43] L. Le, A. Patterson, and M. White, “Supervised autoencoders: Improving generalization performance with unsupervised regularizers,” in *Advances in Neural Information Processing Systems 31*, 2018, pp. 107–117.

[44] A. Ramezani-Kebrya, A. Khisti, , and B. Liang, “Stability of stochastic gradient method with momentum for strongly convex loss functions,” 2019. [Online]. Available: https://openreview.net/forum?id=S1lwRjR9YX

[45] J. Lin, “Divergence measures based on the shannon entropy,” *IEEE Transactions on Information Theory*, vol. 37, no. 1, pp. 145–151, June 1991.

[46] A. Vedaldi and K. Lenc, “Matconvnet: Convolutional neural networks for matlab,” in *ACM International Conference on Multimedia*, 2015.

[47] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” in *International Conference on Learning Representations*, 2015.

[48] O. M. Parkhi, A. Vedaldi, and A. Zisserman, “Deep face recognition,” in *British Machine Vision Conference*, 2015.

[49] R. Rothe, R. Timofte, and L. Van Gool, “Deep expectation of real and apparent age from a single image without facial landmarks,” *International Journal of Computer Vision*, vol. 126, no. 2, pp. 144–157, Apr 2018.

[50] F. Österreicher, “Csiszár’s f-divergence-basic properties,” Victoria University, Melbourne, VIC, Australia, Tech. Rep., 2002.

[51] X. Geng, “Label distribution learning,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 7, pp. 1734–1748, July 2016.

[52] S. Moschoglou, A. Papaioannou, C. Sagonas, J. Deng, I. Kotsia, and S. Zafeiriou, “Agedb: The first manually collected, in-the-wild age database,” in *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, July 2017, pp. 1997–2005.

[53] Z. Zhang, Y. Song, and H. Qi, “Age progression/regression by conditional adversarial autoencoder,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017, pp. 4352–4360.

[54] H. Han, C. Otto, X. Liu, and A. K. Jain, “Demographic estimation from face images: Human vs. machine performance,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 6, pp. 1148–1161, June 2015.

[55] K. Ricanek and T. Tesafaye, “Morph: a longitudinal image database of normal adult age-progression,” in *International Conference on Automatic Face and Gesture Recognition (FGR)*, Apr 2006, pp. 341–345.

[56] G. Panis, A. Lanitis, N. Tsapatsoulis, and T. F. Cootes, “Overview of research on facial ageing using the fg-net ageing database,” *IET Biometrics*, vol. 5, no. 2, pp. 37–46, May 2016.

[57] N. C. Ebner, M. Riediger, and U. Lindenberger, “Faces—a database of facial expressions in young, middle-aged, and older women and men: Development and validation,” *Behavior Research Methods*, vol. 42, no. 1, pp. 351–362, Feb 2010.

[58] M. Grgic, K. Delac, and S. Grgic, “Scface – surveillance cameras face database,” *Multimedia Tools and Applications*, vol. 51, no. 3, pp. 863–879, Feb 2011.

[59] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, “Joint face detection and alignment using multitask cascaded convolutional networks,” *IEEE Signal Processing Letters*, vol. 23, no. 10, pp. 1499–1503, Oct 2016.

[60] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, “A discriminative feature learning approach for deep face recognition,” in *European Conference on Computer Vision (ECCV)*. Springer International Publishing, 2016, pp. 499–515.

[61] G. Guo, , Y. Fu, and T. S. Huang, “Human age estimation using bio-inspired features,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2009, pp. 112–119.

[62] B. Doosti, “Hand pose estimation: A survey,” *CoRR*, vol. abs/1903.01013, 2019.

[63] M. Köstinger, P. Wohlhart, P. M. Roth, and H. Bischof, “Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization,” in *IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, 2011, pp. 2144–2151.

[64] N. Gourier, D. Hall, and J. L. Crowley, “Estimating face orientation from robust detection of salient facial structures,” in *FG Net Workshop on Visual Observation of Deictic Gestures*, 2004.

[65] J. A. Black, M. Gargesha, K. Kahol, P. Kuchi, and S. Panchanathan, “International conference on information technologies and communications (icitc),” in *A framework for performance evaluation of face recognition algorithms*, 2002, pp. 163–174.

[66] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker, “Multi-pie,” in *IEEE International Conference on Automatic Face Gesture Recognition*, Sep 2008, pp. 1–8.

[67] G. Fanelli, M. Dantone, J. Gall, A. Fossati, and L. Van Gool, “Random forests for real time 3d face analysis,” *Int. J. Comput. Vision*, vol. 101, no. 3, pp. 437–458, February 2013.

[68] K. Sheng, W. Dong, M. Chai, G. Wang, P. Zhou, F. Huang, B.-G. Hu, R. Ji, and C. Ma, “Revisiting image aesthetic assessment via self-supervised feature learning,” *CoRR*, vol. abs/1911.11419, 2020.

[69] X. Jin, L. Wu, X. Li, S. Chen, S. Peng, J. Chi, S. Ge, C. Song, and G. Zhao, “Predicting aesthetic score distribution through cumulative jensen-shannon divergence,” in *AAAI Conference on Artificial Intelligence*, 2018.

[70] V. Hosu, B. Goldlücke, and D. Saupe, "Effective aesthetics prediction with multi-level spatially pooled features," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 9367–9375.

[71] H. Talebi and P. Milanfar, "Nima: Neural image assessment," *IEEE Transactions on Image Processing*, vol. 27, no. 8, pp. 3998–4011, 2018.

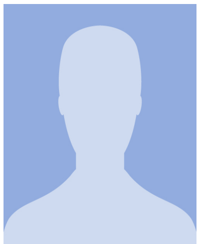
[72] P. Zhao and Z.-H. Zhou, "Label distribution learning by optimal transport," in *AAAI Conference on Artificial Intelligence*, 2018.

[73] S. Kong, X. Shen, Z. Lin, R. Mech, and C. Fowlkes, "Photo aesthetics ranking network with attributes and content adaptation," in *European Conference on Computer Vision (ECCV)*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. Cham: Springer International Publishing, 2016, pp. 662–679.

[74] J. Ren, X. Shen, Z. Lin, R. Mech, and D. J. Foran, "Personalized image aesthetics," in *IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 638–647.

[75] N. Murray, L. Marchesotti, and F. Perronnin, "Ava: A large-scale database for aesthetic visual analysis," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2012, pp. 2408–2415.

Ali Akbari (M'15) received the M.S. degree in Electrical Engineering from the Shiraz University of Technology, Iran in 2012 and the PhD degree in Telecommunications at the Institut supérieur d'électronique de Paris (ISEP), Sorbonne Université, Paris, France in March 2018. Since July 2018, he joined the Centre for Vision, Speech and Signal Processing (CVSSP), University of Surrey, UK as a research fellow to enrich his experiences in the field of face recognition. His research interests include image and video processing, computer vision, deep learning and dictionary learning.



Muhammad Awais received the B.Sc. degree in mathematics and physics from the AJK University in 2001, B.Sc. degree in computer engineering from UET Taxila in 2005, M.Sc in signal processing and machine intelligence and PhD in machine learning from the University of Surrey in 2008 and 2011. He is currently a senior research fellow at the Centre for Vision, Speech and Signal Processing (CVSSP) at the University of Surrey. His research interests include machine learning, deep learning, self(un,semi)-supervised learning, NLP, audio-visual analysis, medical image analysis and computer vision.

Manijeh Bashar (S'16–M'20) received the B.Sc. degree in electrical engineering from the University of Guilan, Iran, in 2009, and the M.Sc. degree in communication systems engineering (with honors) from the Shiraz University of Technology, Iran, in 2013. She received the Ph.D. degree in communications engineering from the University of York, U.K. in 2019. In 2017, she was an Academic Visitor with the Department of Electronics and Nanoengineering, Aalto University, Espoo, Finland, with a Short Term Scientific Mission (STSM) Scholarship Award from European COST-IC1004 "Cooperative Radio Communications for Green Smart Environments".

She is currently a research fellow at the Institute for Communication Systems (ICS), the Home of 5G Innovation Centre (5GIC), University of Surrey. Her current research interests include cooperative communications for 5G networks including distributed massive MIMO, Cloud-RAN, Fog-RAN, NOMA, deep learning, resource allocation, and also millimetre-wave channel modelling.

She received the K. M. Stott Prize for excellent Ph.D. research in electronics engineering from the University of York, U.K. in 2019. She has been awarded First place (based on jury) in the IEEE WCNC'18 three-minute Ph.D. thesis competition for her research in cell-free massive MIMO. She has been nominated for Departmental Prize for Excellence in Research in 2019 at the University of Surrey. She has been a member of Technical Program Committees (TPC) for the IEEE ICC 2020.

Josef Kittler (M'74-LM'12) received the B.A., Ph.D., and D.Sc. degrees from the University of Cambridge, in 1971, 1974, and 1991, respectively. He is a distinguished Professor of Machine Intelligence at the Centre for Vision, Speech and Signal Processing, University of Surrey, Guildford, U.K. He conducts research in biometrics, video and image dataset retrieval, medical image analysis, and cognitive vision. He published the textbook Pattern Recognition: A Statistical Approach and over 700 scientific papers. His publications have been cited more than 68,000 times (Google Scholar).

He is series editor of Springer Lecture Notes on Computer Science. He currently serves on the Editorial Boards of Pattern Recognition Letters, Pattern Recognition and Artificial Intelligence, Pattern Analysis and Applications. He also served as a member of the Editorial Board of IEEE Transactions on Pattern Analysis and Machine Intelligence during 1982-1985. He served on the Governing Board of the International Association for Pattern Recognition (IAPR) as one of the two British representatives during the period 1982-2005, President of the IAPR during 1994-1996. Currently he is a member of the KS Fu Prize Committee of IAPR.