Statistical independence measure based on maximum norm of joint and product-marginal characteristic functions

povilas.daniusis, povilasd@neurotechnology.com

November 2021

Abstract

In this paper we propose statistical independence measure based on the maximum norm of difference between joint and product-marginal characteristic functions, and its estimation procedure. We also extend the proposed measure to the reproducing kernel Hilbert spaces (RKHS).

On the empirical side, we discuss simulated examples, and applications for feature selection/extraction, regularisation, and conduct corresponding experiments with diverse collection of data sets from different domains.

1 Introduction

Statistical dependence measures plays important role in various statistical and machine learning methods (e.g. hypothesis testing [1], feature selection and extraction [2], information bottleneck methods [3], causal inference [4], self-supervised learning [5], causal inference [4], among others). Earliest statistical dependence estimation ideas (e.g. conditional probability) share nearly-common origin with the beginning of formal statistical reasoning itself. During last two centuries ideas of correlation and (relative) entropy (including various generalizations) were proposed and became very popular in numerous applications and theoretical developments. However, with the increasing popularity of statistical machine learning, new statistical dependence estimation methods, that are robust, applicable to noisy, high-dimensional, structured data, and which can be efficiently integrated with modern machine learning methods are helpful for the development both of the theory and application.

In this article we focus on quantitative estimation of statistical independence, using characteristic functions. We begin with the short review of some important previous dependence estimation approaches (Section 2), devoting special attention to ones based on characteristic functions (Section 2.1). Afterwards, we formulate the proposed characteristic function-based statistical dependence measure and its empirical estimator (Section 3), including an extension into reproducing kernel Hilbert spaces (RKHS'es), which are the main theoretical contribution of our paper. Section 4 is devoted to experiments with simulated and real data sets, where we apply the proposed dependence measure for feature extraction and deep neural network (DNN) regularisation, and finalizing Section 5 concludes this article.

2 Previous work

Shannon mutual information [6] and generalizations [7], Hilbert-Schmidt independence criterion [1] and generalizatios [**?**], [8] copula-based kernel dependence measures.

2.1 Characteristic-function-based methods

Characteristic function of d_X-dimensional random vector X defined in some probability space $(\Omega_X, \Sigma_X, \mathbb{P}_X)$ is defined as

$$\phi_X(\alpha) = \mathbb{E}_X e^{i\alpha^T X}, \tag{1}$$

where $i = \sqrt{-1}$, $\alpha \in R^{d_X}$. Having n i.i.d. realisations of X, corresponding empirical characteristic function is defined as

$$\widehat{\phi_X}(\alpha) = \frac{1}{n}\sum_{j=1}^{n} e^{i<\alpha, x_j>}. \tag{2}$$

Having pair of two random vectors (X, Y) defined in another probability space $(\Omega_{X,Y}, \Sigma_{X,Y}, \mathbb{P}_{X,Y})$ joint characteristic function is defined as:

$$\phi_{X,Y}(\alpha, \beta) = \mathbb{E}_{X,Y} e^{i(\alpha^T X + \beta^T Y)}, \tag{3}$$

where $\alpha \in R^{d_X}$ and $\beta \in R^{d_Y}$. Similarly, having n i.i.d. realisations of (X, Y), joint empirical characteristic function is defined as

$$\widehat{\phi_{X,Y}}(\alpha, \beta) = \frac{1}{n}\sum_{j=1}^{n} e^{i(<\alpha, x_j> + <\beta, y_j>)}. \tag{4}$$

In terms of characteristic functions, statistical independence of X and Y is equivalent to $\forall \alpha \in R^{d_x}, \forall \beta \in R^{d_y}$,

$$\Delta_{X,Y}(\alpha, \beta) := \phi_{X,Y}(\alpha, \beta) - \phi_X(\alpha)\phi_Y(\beta) = 0, \tag{5}$$

where d_x and d_y are dimensions of X and Y, respectively.

This formulation of statistical independence was used as the basis (first in [9] for univariate case, and afterwards extended and developed by [10] for bivariate multidimensional random vectors) for construction of statistical independence tests and measures.*Distance covariance* and *distance correlation*, prosposed by

[10] relies on weighted L^2-norm analysis of (5). They select weighting function in such a way, that dependence measure can be expressed in terms of correlection of data-dependent distances. Study [11] generalises [10] to multivariable case and proposes *distance multivariance* and derivative dependence measure, called *total distance multivariance.*

Our motivation stems from the fact that evaluation of [10] measures in high dimensional cases may be prone to curse of dimensionality (as mentioned in [12]). Although staying in L^p-space framework, insted of $p = 2$ (L^2 space) we take a limit when $p \to \infty$, and also avoid direct calculation of integral by working in L^∞, which has corresponding supremum norm. Also, from practical point of view maximization is convenient, because it is already implemented in various deep learning frameworks.

3 Proposed Independence Measure

This motivates the construction of a novel dependence measure, which we further refer to as Kac independence measure (KacIM). Let X and Y be two standartized random random vectors. The proposed independence measure is defined as

$$\kappa(X,Y) = \max_{\alpha \in \mathbb{R}^{d_X}, \beta \in \mathbb{R}^{d_Y}} |\phi_{X,Y}(\alpha,\beta) - \phi_X(\alpha)\phi_Y(\beta)|. \tag{6}$$

3.1 Properties

Theorem 1. *Statistical independence measure* (6) *has the following properties:*

- $\kappa(X,Y) = \kappa(Y,X)$,
- $0 \le \kappa(X,Y) \le 1$,
- $\kappa(X,Y) = 0$ *iff* $X \perp Y$.

Proof. Proof Symmetry is obvious from definition (6) (commutativity of addition and multiplication), and second property directly follows from Cauchy inequality and that absolute value of characteristic function is bounded by 1:

$$|\phi_{X,Y}(\alpha,\beta) - \phi_X(\alpha)\phi_Y(\beta)|^2 = \mathbb{E}_{X,Y}|(e^{i\alpha^T X} - \phi_X(\alpha))(e^{i\beta^T Y} - \phi_Y(\beta))|^2 =$$

$$\mathbb{E}_{X,Y}|(e^{i\alpha^T X} - \phi_X(\alpha))|^2|(e^{i\beta^T Y} - \phi_Y(\beta))|^2 = (1 - |\phi_X(\alpha)|^2)(1 - |\phi_Y(\beta)|^2).$$

□

3.2 Estimation

Having i.i.d. standartized data (x_j, y_j), $j = 1, 2, ..., n$, an empirical scale-invariant estimator of (6) is defined via corresponding empirical characteristic functions (4) and (2):

$$\hat{\kappa}(X,Y) = \max_{\|\alpha\|=\|\beta\|=1} |\widehat{\phi_{X,Y}}(\alpha,\beta) - \widehat{\phi_X}(\alpha)\widehat{\phi_Y}(\beta)|. \tag{7}$$

Empirical estimator (7) also is symmetric and and bounded (Theorem 1) . Normalisation of parameters α and β on to unit sphere is included due to stability issues (really this is the reason?). The estimator (7) can be calculated by using Algorithm 1 [1].

Algorithm 1 KacIM estimator computation algorithm

Require: data batch (x, y), gradient-based optimiser $GradOpt(loss)$

Normalize (x, y) to zero mean and unit variance (scale invariance).

Calculate KacIM estimator $\hat{\kappa}(x, y)$, without maximization step (i.e. using current α, β).

Perform one maximization iteration of computed $\hat{\kappa}(x, y)$ via $\alpha, \beta \rightarrow GradOpt(\hat{\kappa}(x, y))$.

In our implementation we use decoupled weight decay regularization optimizer [14].

3.3 Kernel version

Having two RKHS'es, defined by feature mapping $(x, y) \rightarrow (l(x, .), l(y, .)$, where $k : \mathbb{R}^{d_x} \times \mathbb{R}^{d_x} \rightarrow \mathbb{R}$ and $l : \mathbb{R}^{d_y} \times \mathbb{R}^{d_y} \rightarrow \mathbb{R}$ (see [15]). Then, estimation of kernel-KIM ($\hat{\kappa}_{k,l}(X, Y)$) can be reformulated as maximization of :

$$|\frac{1}{n}\sum_{j=1}^{n} e^{i(<\alpha,k(x_j,.)>+<\beta,l(y_j>,.))} - \frac{1}{n^2}\sum_{j=1}^{n} e^{i<\alpha,k(x_j,.)>}\sum_{k=1}^{n} e^{i<\beta,l(y_k,.)>}|, \tag{8}$$

and representer theorem[**?**] implies

$$\hat{\kappa}_{k,l}(X, Y) = \max_{\|\alpha\|=\|\beta\|=1} |\frac{1}{n} 1^T e^{i(\alpha^T K + \beta^T L)} - \frac{1}{n^2}(1^T e^{i(\alpha^T K)})(1^T e^{i(\beta^T L)})|, \tag{9}$$

where K and L are Gram matrices, corresponding to x_i and y_i. Note that the number of parameters of $\hat{\kappa}_{k,l}(X, Y)$ is dimension-idnependent and is equal to $2n_b$, where n_b is batch size. Also, kernel-KIM can be applied for structured data, via corresponding positive defined kernels.

4 Experiments

Further we will conduct empirical investigation of KacIM. We will begin with simple illustrative simulations, and afterwards investigate empirical performance of KacIM in classifier feature extraction and regularization tasks using various publicly available data sets.

[1] Pytorch [13] implementationcan be accessed from `https://github.com/povidanius/kac_independence_measure`

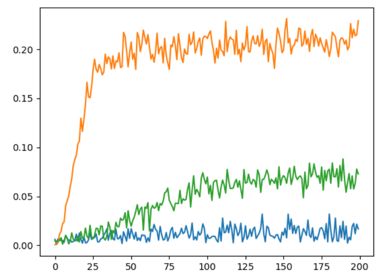

Figure 1: Dependence detection in independent data (blue), additive (orange) and multiplicative (green) noise scenarios.

4.1 Generated data

Non-linear statistical dependence detection We begin with simple example, which demonstrates the efficiency of KacIM for simulated multivariate data with additive and multiplicative noise.

In Figure 4.1 reflects KacIM values during iterative adaptation (200 iterations). In the case of independent data, both x_i and y_i ($d_x = 512$, $d_y = 4$) are sampled from gaussian distribution, independently. In the case of dependent data, an additive noise and multiplicative noise, the dependent variable is generated according to $y_i = sin(Px_i) + cos(Px_i) + \lambda\epsilon_i$ ($\lambda = 1.00$) and $y_i = (sin(Px_i) + cos(Px_i))\epsilon_i$, respectively, where P is $d_x \times d_y$ random projection matrix, $\epsilon_i \sim N(0, 1)$ and $\epsilon_i \perp x_i$.

When data is independent, both in additive and multiplicative cases, due to independence, estimator (7) is resistant to maximisation, and oscillates near zero. On the other hand, when the data is not independent, the condition (5) is violated and maximization of estimator (7) is possible.

Noise variance effect In this simulation we use the same additive noise setting as in previous paragraph, but evaluate all noise levels $\lambda \in [0.1, 3.0]$, with step 0.1. Figure 4.1 empirically shows that value of KacIM negatively correlates with noise level, and therefore the proposed measure is able not only to detect whether independence is present, but also to quantitatively evaluate it, which enables to use it to derive cost functions for various learning-based algorithms.

Since in 1 we standartize data, KacIM is also scale-invariant (i.e. $\kappa(rx, ry) = \kappa(x, y)$), where $r > 0$ is scale parameter.

4.2 Feature Extraction

In feature extraction experiments we will use a set of classification data sets from OpenML [16]. The purpose of these experiments is to provide the preliminary evaluation of the applicability of KacIM for feature extraction, hence we use

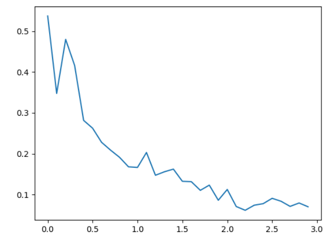

Figure 2: Noise level (x axis) vs final iteration KacIM value (y axis). KacIM values for larger noise levels saturates as in tail of graph.

Classification accuracies					
	size/dim.	KNN(3)	LR	LSVM	QSVM
Ionosphere	(351, 34)	AF	AFG	004	
Spambase	(4601, 57)				
One-hundred-plants-texture	(1599, 64)	AF	AFG	004	
LSVT	(126, 310)	AF	AFG	004	
Micro-mass	(360, 1300)	AF	AFG	004	
Tokyo1	(959, 44)	AF	AFG	004	
Clean1	(476, 168)	AF	AFG	004	

Table 1: KNN

rather basic comparative baselines. We conduct linear feature extraction by seeking

$$W^* = arg\max_W \kappa(Wx, y) - \alpha Tr\{(W^TW - I)^T(W^TW - I)\}, \qquad (10)$$

where the regularisation term, controlled by multiplier $\alpha \geq 0$, enforces projection matrix W^* to be orthogonal.
Afterwards, feature extraction is conducted by $f = W^*x$ and these features are used as the inputs to several popular classifiers: 3-nearest neighbor classifier with Euclidean distance (KNN(3)), logistic regression (LR), linear and quadratic support vector machine[?] (LSVM and QSVM, respectively). We randomly split all the datasets in training and testing sets of equal size, comparing unmodified inputs x, and features of all possible dimensions up to d_x. In our experiments we set α to 1.0 to quickly ensure orthogonal projection matrices, and further proceed to dependence maximization stage. The classification accuracies, reported in Table 4.2 demonstrate that this KacIM-based feature extraction procedure indeed allows to increase classification accuracy when applied to real data sets from various domains.

4.3 Regularisation

In regularisation experiments we investigate chest x-ray classification task. It is represented as binary classification data set, consisting of xray scans, which should be classified as pneumonia or normal.

Let $f(x|\theta)$ be DNN classifier. We will investigate additive regularizer, which maximises depency of bottleneck feature $\phi(x)$ and target variable y:

$$Cost(\theta) = CE(f(x|\theta), y) + \beta\kappa(\phi(x|\theta), y), \tag{11}$$

where CE is cross-entropy loss, and $\beta \geq 0$ is regularisation parameters.

5 Discussion

In this article we propose statistical dependence measure, KacIM, which relies on the L^∞ norm of the difference between joint characteristic function and the product of marginal ones.

Although we formulated and analysed KacIM for bivariate vectorial case, similarly it can be generalised for multivariate case. In addition, since characteristic functions are defined for matrices, graphs, and other objects [**?**], the proposed dependence measure can be extended to those objects as well, which is potential direction of future research of KacIM.

Conducted empirical analysis show, that KacIM can detect and measure statistical independence for non-linearly related, high-dimensional data, and that it can be applied for feature extraction and DNN regularisation and improve model's performance on real data sets. On the other side, direct applications of the proposed statistical dependence measure in the areas of causality, information bottleneck, and other domains where not explored in this study, and left for the future work, as well as its important properties (e.g. interpretability, limit distribution).

6 Acknowledgements

References

[1] Arthur Gretton, Olivier Bousquet, Alex Smola, and Bernhard Schölkopf. Measuring statistical dependence with hilbert-schmidt norms. In *ALT*, 2005.

[2] P. Daniušis and P. Vaitkus. Supervised feature extraction using hilbert-schmidt norms. In Emilio Corchado and Hujun Yin, editors, *Intelligent Data Engineering and Automated Learning - IDEAL 2009*, pages 25–33, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

[3] Kurt Wan-Duo Ma, J. P. Lewis, and W. Kleijn. The hsic bottleneck: Deep learning without back-propagation. *ArXiv*, abs/1908.01580, 2020.

[4] Patrik Hoyer, Dominik Janzing, Joris M Mooij, Jonas Peters, and Bernhard Schölkopf. Nonlinear causal discovery with additive noise models. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, editors, *Advances in Neural Information Processing Systems*, volume 21. Curran Associates, Inc., 2009.

[5] Yazhe Li, Roman Pogodin, Danica J. Sutherland, and Arthur Gretton. Self-supervised learning with kernel dependence maximization. In *Thirty-Fifth Conference on Neural Information Processing Systems*, 2021.

[6] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory 2nd Edition (Wiley Series in Telecommunications and Signal Processing)*. Wiley-Interscience, July 2006.

[7] José M. Amigó, Sámuel G. Balogh, and Sergio Hernández. A brief review of generalized entropies. *Entropy*, 20(11), 2018.

[8] Barnabás Póczos, Zoubin Ghahramani, and Jeff G. Schneider. Copula-based kernel dependency measures. *ArXiv*, abs/1206.4682, 2012.

[9] Andrey Feuerverger. A consistent test for bivariate dependence. *International Statistical Review / Revue Internationale de Statistique*, 61(3):419–433, 1993.

[10] Gábor J. Székely, Maria L. Rizzo, and Nail K. Bakirov. Measuring and testing dependence by correlation of distances. *The Annals of Statistics*, 35(6):2769 – 2794, 2007.

[11] Björn Böttcher, Martin Keller-Ressel, and René Schilling. Distance multivariance: New dependence measures for random vectors, 10 2018.

[12] Dominic Edelmann, Konstantinos Fokianos, and Maria Pitsillou. An Updated Literature Review of Distance Correlation and Its Applications to Time Series. *International Statistical Review*, 87(2):237–262, August 2019.

[13] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.

[14] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019.

[15] Bernhard Schlkopf, Alexander J. Smola, and Francis Bach. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. The MIT Press, 2018.

[16] Joaquin Vanschoren, Jan N. van Rijn, Bernd Bischl, and Luis Torgo. Openml: Networked science in machine learning. *SIGKDD Explorations*, 15(2):49–60, 2013.