Statistical independence measure based on maximum norm and characteristic function factorisation

Povilas Daniušis, povilasd@neurotechnology.com*

November 2021

Abstract

In this paper we propose statistical independence measure based on the maximum norm of the absolute value of difference between joint and product-marginal characteristic functions, and its estimation procedure (including open-source repository). We also extend the proposed measure to the reproducing kernel Hilbert spaces (RKHS), which allows to apply it to structured data.

We conduct experiments both with simulated and real data. Our experiments reveal, that the proposed measure can exploit statistical dependence in non-linear data sets, and that it can improve real-data classification accuracy, when applied for feature extraction and regularisation.

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, 3], information bottleneck methods [4], causal inference [5], self-supervised learning [6], representation learning [7], 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

[*] This preprint currently is not officially related to Neurotechnology

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

During recent years, various approaches have been used in order to construct statistical dependence estimation methods. For example, information theory (mutual information [8] and generalisations), reproducing kernel Hilbert spaces (Hilbert-Schmidt independence criterion [1]), characteristic functions (distance correlation [9, 10]), and other (e.g. [11] copula-based kernel dependence measures, integral-porbability-metric-reliant Sobolev independence criterion [12]). Further we will focus on characteristic-function-based methods.

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

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

If cumulative distribution function (cdf.) of (X, Y), $F_{X,Y}(x, y)$, $x \in R^{d_X}$ and $y \in R^{d_Y}$ factorises as $F_X(x)F_Y(y)$ for all x and y, X and Y are called independent (the same holds for probability density function, pdf.). However, this criterion is impractical due to need of evaluation of potentially high-dimensional cdf. or pdf., and often alternative independence criterions are more useful. For example, in terms of characteristic functions, statistical independence of X and Y is equivalent to $\forall \alpha \in R^{d_X}, \forall \beta \in R^{d_Y}$,

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

This formulation of statistical independence was used as the basis (first in [9] for one-dimensional 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*, prospsoed 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 [13] generalises [10] to multivariable case and proposes *distance multivariance* and derivative dependence measure, called *total distance multivariance.* [14] proposed computationally efficient algorithm for estimation of distance correlation measure, reducing computational complexity from $O(n^2)$ to $O(n \cdot \log n)$, where n is sample size.

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 [15]). Although staying in L^p-space framework, instead of $p = 2$ (L^2 space) we take a limit when $p \to \infty$, and thereby avoid direct calculation of norm integral, since norm of L^p converges to supremum norm when $p \to \infty$. Also, from practical point of view maximization is convenient, because it is efficiently implemented in modern deep learning frameworks (e.g. Pytorch [16]).

3 Proposed Independence Measure

The above considerations serves as the basis for constructing 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 Basic Properties

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

1. $\kappa(X, Y) = \kappa(Y, X)$,
2. $0 \leq \kappa(X, Y) \leq 1$,
3. $\kappa(X, Y) = 0$ *iff* $X \perp Y$.
4. $\kappa(X, Y)$ *is scale invariant.*

Proof. Property *1.* is obvious from definition (6) (commutativity of addition and multiplication), and property *2.* directly follows from Cauchy inequality and that absolute value of characteristic function is bounded by 1:

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

Proof of property *3.* directly follows from properties of characteristic functions (see e.g. [17], Corollary 14.1)[1]. Scale invariance (Property *4.*) is trivial result of the standartisation requirement for X and Y. □

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):

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

Empirical estimator (7) also is symmetric and and bounded (Theorem 1). It can be calculated iteratively by Algorithm 1 (Pytorch [16] implementation can be accessed from `https://github.com/povidanius/kac_independence_measure`).

Algorithm 1 KacIM estimation iteration

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

Algorithm 1 requires to initialise α and β (we empirically found that uniform initialisation resulted in faster convergence), select stoping criteria (e.g. $k \in \mathbf{N}$), and optimiser. In our implementation we use decoupled weight decay regularization optimizer [19]. We also emprically observed that normalisation of parameters α and β on to unit sphere increases estimation stability (why?).After the estimation of KacIM via Algorithm 1, the evaluation the estimator (7) has computation complexity $O(n)$, where n is sample size.

Note, that e.g. Shannon and Renyi mutual information [8] are also estimated via transforming them into maximisation problem by Donsker-Varadhan representation [20], in order to avoid density estimation.

[1] This property also is known as Kac's theorem [18]. Although it is quite simple mathematical fact, this provides the basis of the proposed measure's name.

3.3 Connection to canonical correlation

If both X and Y are zero mean Gaussian random vectors we have:

$$\kappa(X, Y) = \max_{\alpha,\beta} |e^{-\frac{1}{2}(\alpha^T \Sigma_x \alpha + \beta^T \Sigma_y \beta)}(e^{-\frac{1}{2}\alpha^T \Sigma_{x,y} \beta} - 1)|. \tag{8}$$

Assuming constant $\alpha^T \Sigma_x \alpha$ and $\beta^T \Sigma_y \beta$, the maximization of (8) can be achieved by maximization of $\alpha^T \Sigma_{x,y} \beta$, which coincides with the cost function of canonical correlation analysis [**?**]. Here Σ_x, Σ_y and $\Sigma_{x,y}$ are covariance matrices of X, Y, and (X, Y), respectively.

3.4 Kernel version

Having two RKHS'es, defined by feature maps $k, l : (x, y) \to (k(x, .), l(y, .)$, where $k : \mathbb{R}^{d_x} \times \mathbb{R}^{d_x} \to \mathbb{R}$ and $l : \mathbb{R}^{d_y} \times \mathbb{R}^{d_y} \to \mathbb{R}$ (see [21]). Then, estimation of kernel-KIM ($\hat{\kappa}_{k,l}(X, Y)$) can be reformulated as maximization of :

$$|\frac{1}{n}\sum_{j=1}^{n} e^{i(<\alpha,k(x_j,.)>+<\beta,l(y_j>,.))} - \frac{1}{n^2}\sum_{j=1}^{n} e^{i<\alpha,k(x_j,.)>}\sum_{k=1}^{n} e^{i<\beta,l(y_k,.)>}|, \tag{9}$$

and representer theorem[**?**] implies

$$\hat{\kappa}_{k,l}(X, Y) = \max_{\|\alpha\|=\|\beta\|=1} |\frac{1}{n}1^T e^{i(\alpha^T K + \beta^T L)} - \frac{1}{n^2}(1^T e^{i(\alpha^T K)})(1^T e^{i(\beta^T L)})|, \tag{10}$$

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 in order to demonstrate that it can measure statistical dependencies in non-linear data sets, and that it can be practically useful as a component of cost functions (in feature selection and extraction, and regularisation problems).

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.

Figure 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

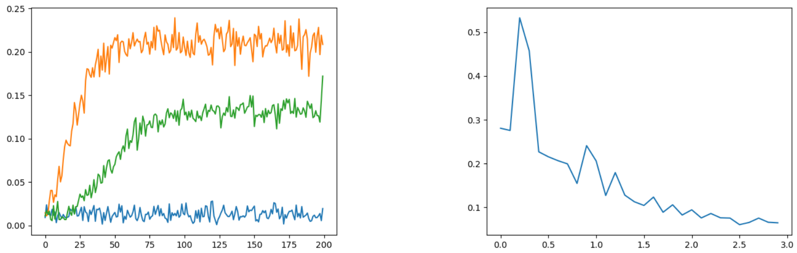

Figure 1: Left figure: Dependence detection in independent data (blue), additive (orange) and multiplicative (green) noise scenarios. Righ figure: noise level (x axis) vs final iteration KacIM value (y axis). KacIM values for larger noise levels saturates as in tail of graph

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 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.

Comparison with distance correlation

4.2 Feature extraction

Let use denote by $T := (x_i, y_i)_{i=1}^N$ a supervised-learning dataset of N pairs of d_x-dimensional inputs x_i, and d_y-dimensional one-hot-encoded outputs y_i.

In feature extraction experiments we will use a set of classification data sets from OpenML [22], which cover different domains. We use $KacIM$ in order to conduct supervised linear feature extraction by seeking

$$W^* = arg\max_W \kappa(Wx, y) - \alpha Tr\{(W^TW - I)^T(W^TW - I)\}, \tag{11}$$

Dataset	$N/d_x/n_c$.	Raw	KacIMFE	NCA
isolet	(7797,617,26)	0.9261	0.9437	**0.9477**
madelon	(2600,500,2)	**0.6015**	0.5484	0.5685
prnn-viruses	(61,18,4)	0.6452	0.9265	0.9355
ionosphere	(351,34,2)	0.8807	0.9278	**0.9375**
micro-mass	(360,1300,10)	0.8778	**0.9282**	0.8944
clean1	(476,168,2)	0.7689	**0.9888**	0.9790
robot-failures-lp2	(47,90,5)	0.4583	0.6067	0.5833
waveform-5000	(5000,40,3)	**0.8692**	0.8017	0.8516
spambase	(4601,57,2)	0.6906	0.8285	**0.8705**
gina-agnostic	(3468,970,2)	**0.8512**	0.7894	0.8080
scene	(2407,299,2)	0.8895	**0.9707**	0.9336
tokyo1	(959,44,2)	0.7250	0.8995	**0.9062**
one-hundred-plants-shape	(1600,64,100)	0.1013	**0.4913**	0.4688

Table 1: Classification accuracies. N denotes full data set size, d_x - input dimensionality, and n_c - number of classes. In this table feature dimension is equal to a half of original input dimension. Best accuracies that are also statistically significant (Wilcoxon's signed rank test [24], 25 runs, p-value threshold 0.01) are indicated in bold text.

where the regularisation term, controlled by multiplier $\alpha \geq 0$, enforces orthogonality of projection matrix W^*, and $Tr\{.\}$ denotes matrix trace operator.

In all the experiments (11) the cost function is optimised iteratively (250 iterations), simultaneously optimising parameters of KacIM (α and β) and projection matrix W. After the optimisation, the feature extraction is conducted by $f(x) = W^*x$, where x is original input vector, and f are corresponding feature vector.

We randomly split all the datasets in training and testing sets of equal size. In our experiments we set α to 1.0 to quickly ensure orthogonal projection matrices, and further proceed to dependence maximization stage. In order to quantitatively evaluate features, we use logistic regression-based classification accuracy, measured on the testing set.

We use two baselines: raw features (RAW column in Table 1) and neighborhood component analysis [23] (NCA column in Table 1). The purpose of these experiments is to provide the preliminary evaluation of the applicability of KacIM for feature extraction, hence we use rather basic cost function and comparative baselines.

The classification accuracies, reported in Table 1 demonstrate that KacIM-based feature extraction procedure (KacIMFE column) indeed allows to increase classification accuracy when applied to real data sets from different domains. In contrast to our feature extraction approach, NCA explicitly optimises for classification accuracy, rather than more abstract dependency of features $f(x)$ with dependent variable y.

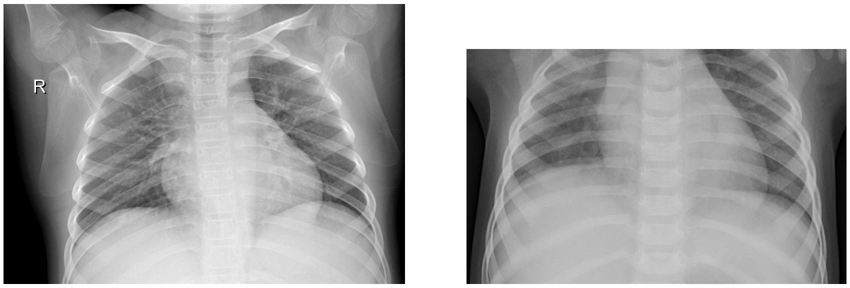

Figure 2: Left figure: example of healthy lung x-ray. Right: pneumonia.

Mode	esting
Without regularisation	0.8338
With regularisation	0.8427

Table 2: Classification accuracy comparison of regularised and not regularised model (pneumonia dataset).

4.3 Regularisation

In regularisation experiments we investigate chest x-ray classification task. It is represented as binary classification data set, consisting of x-ray scans (5216 for training, and 642 for testing), which should be classified as pneumonia or normal (e.g. Figure 2). As classifier we use *ResNet18*, trained with batches of 128 elements. We denote classifier as $f(\phi(x|\theta_0)|\theta_1)$, where θ_0 are bottleneck parameters, θ_1 final (linear) layer parameters, and x is 224×224 input image. For optimisation we use decoupled weight decay regularization optimizer [19] with learning rate set to 0.0002, and weight decay parameter set to 0.00001 (7 epochs). We used the following data augmentations: random horizontal flip, random rotation (up to 10 degrees), color jitter.

We will investigate additive regularizer, which maximises depency of bottleneck the feature $\phi(x)$ and target variable y (one-hot encoding):

$$Cost(\theta_0, \theta_1, W) := CE(f(\phi(x|\theta_0)|\theta_1), y) - \beta\kappa(W\phi(x|\theta_0), y), \tag{12}$$

where $CE(.,.)$ is cross-entropy loss, W is 32×512 projection matrix, and $\beta \geq 0$ is regularisation parameter (in our experiments $\beta = 0.1$).

The results of classification accuracy without and with aforementioned regularisation is reported in Table 2.

5 Conclusion

In this article we propose statistical dependence measure, KacIM, which corresponds to the L^∞ norm of the absolute value of difference between joint characteristic function and the product of marginal ones. The proposed measure, in theory can detect both linear and non-linear statistical depdendence between a pairs of random variables of possibly different dimension, extended to various known statistical generalisations (e.g. reproducing kernel Hilbert spaces, multivariablity), machine learning scenarios (e.g. feature extraction, regularisation, among others), and is empirically tractable on these problems. On the other side, it raises a corresponding set of unanswered questions, both theoretical and empirical. For example, the interpretability when it approaches its maximal value remains insufficiently clear, however empirical experiments with simulated data reveals, that increasing independence between two random variables is reflected in a decreasing trend on the estimated values of the proposed dependence measure. In contrary to e.g. HSIC or distance correlation, the estimation of the porposed measure is iterative optimisation process. Therefore, parameter initialization, meta-parameter (e.g. stopping criteria, batch size) selection are needed in order to evaluate it efficiently.

Beside demonstrated applications in Section 4, the proposed measure is differentiable and thereby can be integrated with various modern deep-learning methods, applied to high-dimensional and structured data. From empirical point of view, we see exploration of KacIM in causality, information bottleneck theory, self-supervised learning, and other modern problems, where dependence measures define a criterion of optimisation, as important future work.

6 Acknowledgements

We sincerely thank Dr. Pranas Vaitkus, Dr. Linas Petkevičius, and colleagues from Neurotechnology for discussions. We also thank Neurotechnology for supporting this research.

References

[1] Arthur Gretton, Olivier Bousquet, Alex Smola, and Bernhard Schölkopf. Measuring Statistical Dependence with Hilbert-Schmidt Norms. In *ALT*, 2005.

[2] Daniušis, P. and Vaitkus, P. Supervised Feature Extraction Using Hilbert-Schmidt Norms. In Corchado, Emilio and Yin, Hujun, editor, *Intelligent Data Engineering and Automated Learning - IDEAL 2009*, pages 25–33, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

[3] P. Daniušis, Pr. Vaitkus, and L. Petkevičius. Hilbert–Schmidt component analysis. *Lithuanian mathematical journal*, 57(A):7–11, Dec. 2016.

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

[5] 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.

[6] 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.

[7] Ruggero Ragonesi, Riccardo Volpi, Jacopo Cavazza, and Vittorio Murino. Learning Unbiased Representations via Mutual Information Backpropagation. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 2723–2732, 2021.

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

[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] Barnabás Póczos, Zoubin Ghahramani, and Jeff G. Schneider. Copula-based Kernel Dependency Measures. *ArXiv*, abs/1206.4682, 2012.

[12] Mroueh, Youssef and Sercu, Tom and Rigotti, Mattia and Padhi, Inkit and Nogueira dos Santos, Cicero. Sobolev Independence Criterion. In *Advances in Neural Information Processing Systems 32, editor = H. Wallach and H. Larochelle and A. Beygelzimer and F. d'Alché-Buc and E. Fox and R. Garnett*, pages 9505–9515. Curran Associates, Inc., 2019.

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

[14] Arin Chaudhuri and Wenhao Hu. A fast algorithm for computing distance correlation. *Computational Statistics and Data Analysis*, 135:15–24, 2019.

[15] 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.

[16] 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.

[17] Jacod Jean. *Probability essentials / Jean Jacod, Philip Protter.* Universitext. Springer, Berlin Heidelberg New York, 2nd edition edition, 2003.

[18] Johan Kustermans J. Martin Lindsay Michael Schuermann Uwe Franz David Applebaum, B.V. Rajarama Bhat. Quantum Independent Increment Processes I: From Classical Probability to Quantum Stochastic Calculus. 2005.

[19] Ilya Loshchilov and Frank Hutter. Decoupled Weight Decay Regularization. In *ICLR*, 2019.

[20] Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeshwar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and Devon Hjelm. Mutual Information Neural Estimation. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 531–540. PMLR, 10–15 Jul 2018.

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

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

[23] Jacob Goldberger, Geoffrey E Hinton, Sam Roweis, and Russ R Salakhutdinov. Neighbourhood Components Analysis. In L. Saul, Y. Weiss, and L. Bottou, editors, *Advances in Neural Information Processing Systems*, volume 17. MIT Press, 2004.

[24] Wilcoxon, Frank. *Individual Comparisons by Ranking Methods*, pages 196–202. Springer New York, New York, NY, 1992.