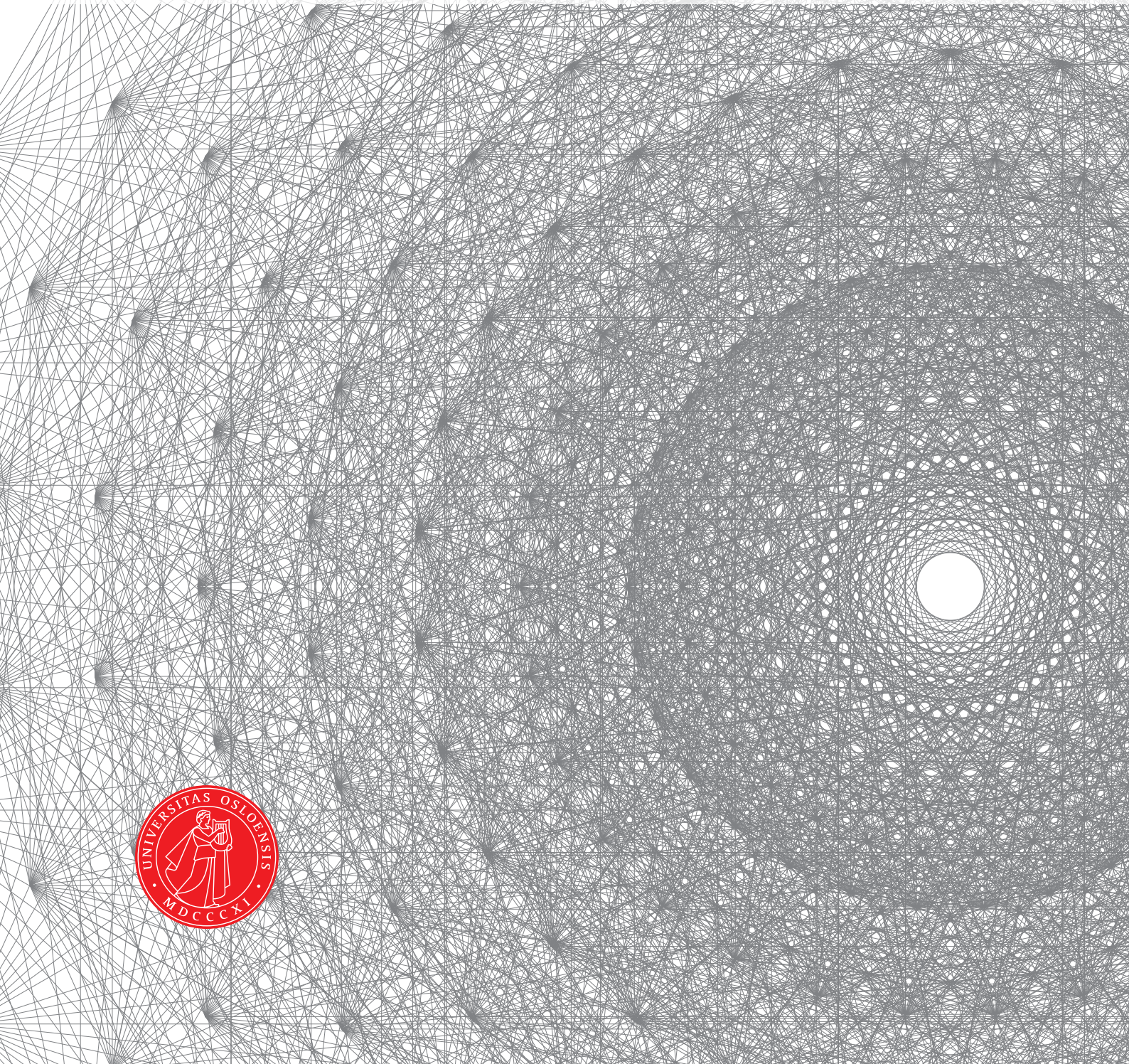
UiO : Department of Mathematics
University of Oslo

# Title

Optional Subtitle

**Erik Lien Bolager**

Master's Thesis, Spring 2021



The study programme is unspecified. Please consult the documentation for the package `masterfrontpage` in order to correctly print the colophon:

https://github.com/martinhelso/masterfrontpage

# CHAPTER 1

# Preliminaries

def: neuralnet_kidger_def

**Definition 1.0.1.** Let $\mathcal{NN}_K^\phi$ be a fully connected neural network where each hidden layer have $K$ nodes, and with activation function $\phi\colon \mathbb{R} \to \mathbb{R}$ that is continuous, nonaffine, and include at least one point that is continuously differentiable with nonzero derivative.

thm: uni_nn

**Theorem 1.0.2** (Kidger et al. 2020)**.** *Let $\mathcal{X} \subset \mathbb{R}^D$ be compact and $\mathcal{NN}_{D+M+2}^\phi$ have input size $D$ and output size $M$. Then, if the network $\mathcal{NN}_{D+M+2}^\phi$ allows for arbitrary depth, is dense in $C(\mathcal{X}; \mathbb{R}^M)$.*

# CHAPTER 2

# Normalizing Flows

## 2.1 Introduction

## 2.2 Base distribution

There are two parts to normalizing flows. The first part is the *base distribution*. In this thesis we do not restrict ourselves to any specific distribution in the following results and discussion, unless explicitly specified. We do however specify three criterias it ought to follow. Let $q_{\boldsymbol{z_0}}$ represent the base distribution over sample space $\mathcal{Z}_0$, then three of its properties ought to be

1. Continuous and parameterized by $\boldsymbol{\theta}$.
2. Sample space $\mathcal{Z}_0$ is defined on a reasonably large subset of $\mathbb{R}^D$.
3. Fast and easy to sample from, as well as evaluating the density $q_{\boldsymbol{z_0}}$.

The second property is there to ensure computationally that we have enough unique points to represent any distribution after transformation. For example, define a distribution with $\mathcal{Z}_0 = (0.999, 0.9999)$. The precision we need in our floating-point representation to represent a distribution of a much wider interval through a bijective transformation is not practical (eventhough the the subset $\mathcal{Z}_0$ has the same cardinality as $\mathbb{R}$ and theoretically is not a problem). Hence, choosing any distribution that fullfill these three properties is adequate. Although, practically there can be benefits of choosing distributions with certain properties e.g $\mathcal{Z}_0 = \mathbb{R}^D$, heavy-tailed, bi-modal, can certainly aid in training w.r.t time, number of transformations needed, and convergence in finite steps.

In particular, any result that is such as convergence or if given flow is an universal density approximator, ought not to depend too much on $q_{\boldsymbol{z_0}}$. This is to make the results as general as possible, and not too assume too much a priori. If not one can trivialize properties such as universal density approximator by simply assuming we have a base distribution that is similar enough to $p_{\boldsymbol{x}}$ such that a simple transformation is adequate. Hence, base distributions such as a multivariate-Gaussian or an Uniform distribution on $[0, 1]$ are ones we typically operate with. In fact, the term "normalizing" comes from the fact that applying the flow backwards from $\boldsymbol{x} \sim q_{\boldsymbol{x}}$, we achieve the base distribution which is often a normal distribution. The exception is that we may limit the space we can sample from to be compact, which in $\mathbb{R}^D$ case is a bounded and closed subset.

We also note that viewing the base distribution as a prior and $q_{\boldsymbol{x}}$ as a posterior, although tempting, is not correct. For starter, fewer data-points does not neccessarily mean a posterior that is closer to the prior, as we can simply overfit the smaller dataset. More crucially, we have no guarantee for $\mathcal{Z}_0 = \mathcal{X}$, hence points with density 0 in "prior" can obtain positive density in "posterior". Hence, normalizing flows, generally, cannot be intepreted as a prior with the flow transforming the prior distribution to the posterior. However, future research may describe a class of flows that follows the description above and may lead to interesting Baysian interpretations.

## 2.3 Flows

Second part of normalizing flows is the transformations of a sample $\boldsymbol{z}_0$ from base density to a different and hopefully more complex distribution. The aim of this section is to define normalzing flow formally, which we start by defining a pushforward measure.

**Definition 2.3.1** (Kobyzev et al. 2020)**.** If $(\mathcal{Z}_0, \Sigma_{\mathcal{Z}_0})$, $(\mathcal{X}, \Sigma_{\mathcal{X}})$ are measurable spaces, $f$ is a measureable mapping between them, and $\mu$ is a measure on $\mathcal{Z}_0$, the one can define a measure on $\mathcal{X}$ as

$$f_*\mu(X) = \mu\left(f^{-1}(X)\right), \text{ for all } X \in \mathcal{X}. \tag{2.1}$$

The measure $f_*\mu(X)$ is known as the *pushforward measure*.

Let $(\mathcal{X}, \Sigma_{\mathcal{X}}, \nu)$ be the measure space we are interested in. Normalizing flows can be seen as a framework that describes classes of functions $f$ and a simpler measure space $(\mathcal{Z}_0, \Sigma_{\mathcal{Z}_0}, \mu)$, such that $f_*\mu = \nu$. When $\mu$ is a probability measure implies that the pushforward measure w.r.t $f$ is also a probability measure. This can easily be proven by the fact that

$$f_*\mu(\mathcal{X}) = \mu\left(f^{-1}(\mathcal{X})\right) = \mu(\mathcal{Z}_0) = 1$$

and by letting $\{X_i \mid i = 1, 2, 3, \dots\}$ be set with pairwise disjoint elements we have

$$\begin{aligned} f_*\mu\left(\bigcup_{i=1}^{\infty} X_i\right) &= \mu\left(\bigcup_{i=1}^{\infty} f^{-1}(X_i)\right) \\ &= \sum_{i=1}^{\infty} \mu\left(f^{-1}(X_i)\right) \\ &= \sum_{i=1}^{\infty} f_*\mu(X_i). \end{aligned}$$

Hence we have countable additivity, which means both the requirements for a probability measure is fulfilled. Normalizing flows can therefore be seen as, starting with a simple probability space $(\mathcal{Z}_0, \Sigma_{\mathcal{Z}_0}, \mu)$, applying $f$ on it to achieve a pushforward distribution $(\mathcal{X}, \Sigma_{\mathcal{X}}, f_*\mu)$. The goal being to find $f$ such that the pushfoward distribution is as close as possible to $(\mathcal{X}, \Sigma_{\mathcal{X}}, \nu)$, w.r.t some divergence measure.

The view above is quite general, and does not lend itself directly to finding an exact density of the pushforward distribution. To achieve this, we need to

constrain the class of functions $f$ and the probability space. Firstly, as we are working with densities, we need to restrain the spaces we are working with such that we are guearanteed that a density exist. For this we need to introduce two properties.

**Definition 2.3.2.** (McDonald et al. 2013) A measure space $(\mathcal{X}, \Sigma_\mathcal{X}, \nu)$ is called a *$\sigma$-finite measure space* if there is a sequence $\{X_n\}_n$ of $\Sigma_\mathcal{X}$-measureable sets such that $\bigcup_n X_n = \mathcal{X}$ and $\nu(X_n) < \infty$ for each $n$.

**Definition 2.3.3.** (McDonald et al. 2013) Let $(\mathcal{X}, \Sigma_\mathcal{X})$ be a measureable space and both $\nu$ and $\lambda$ be measures on $\Sigma_\mathcal{X}$, where $\lambda$ is the Lebesgue measure. Then $\nu$ is *absolutely continuous* with respect to $\lambda$ if $\nu(X) = 0$ whenever $\lambda(X) = 0$. This is also denoted as $\nu \ll \lambda$.

Constraining our measure spaces we are working with to include these two properties, we can guarantee that we have a density function. Any measure space we are working with in future, both base distribution and target distribution, are assumed to follow the properties described above and in the next theorem.

thm: radon

**Theorem 2.3.4** (Radon-Nikodym Theorem)**.** *Let $(\mathcal{X}, \Sigma_\mathcal{X}, \lambda)$ be a $\sigma$-finite measure space and $\nu$ be a $\sigma$-finite measure on $\Sigma_\mathcal{X}$. If $\nu \ll \lambda$ then there is a nonnegative extended real-valued $\Sigma_\mathcal{X}$-measurable function $p_\mathcal{X}$ on $\mathcal{X}$ such that*

$$\nu(X) = \int_X p_\mathcal{X} \, d\lambda, \quad X \in \Sigma_\mathcal{X}. \tag{2.2}$$

Hence the nonnegative function is the density, also called the Radon-Nikodym derivative. In this thesis we shall constrict ourselves to work on $\mathcal{X} \subseteq \mathbb{R}^D$ and with the Borel $\sigma$-algebra denoted $\mathscr{B}(\mathcal{X})$, for both the base and target distribution. The probability spaces based of these sample spaces and $\sigma$-algebras fulfills the conditions descibed in Theorem 2.3.4.

Limiting the function $f$ will be the last necessary component, such that we can evaluate the density of the pushforward measure.

**Definition 2.3.5.** A function $f \colon \mathbb{R}^D \to \mathbb{R}^D$ is a diffeomorphism if it is bijective and both itself and its inverse is differentiable. If $f$ and $f^{-1}$ is $r$ times continuous differentiable, we define it as a $C^r$-diffeomorphism.

Restricting ourselves to $f$ being at least a $C^1$-diffeomorphism is unnecessary and limiting. We therfore define piecewise diffeomorphisms w.r.t some distribution.

**Definition 2.3.6.** Let $(\mathcal{X}, \mathscr{B}(\mathcal{X}), \mu)$ be a probability space with a density $p$. Let $X_i$ for $i = 0, 1, 2, \ldots, k$ be a partion of $\mathcal{X}$ such that $\mu(x \in X_0) = 0$. A piecewise-diffeomorphism $f \colon \mathcal{X} \to \mathbb{R}^D$ w.r.t $p$ is continuous and restricted to $X_i$ is a diffeomorphism. That is, $f_i \colon X_i \to \mathbb{R}^D$ is a diffeomorphism, for all $i = 1, 2, \ldots, k$. When all $f_i$'s are $C^r$-diffeomorphisms makes $f$ a piecewise $C^r$-diffeomorphism w.r.t $p$.

Hence, we shall restrict our choices of $f$ to be $C^1_p$-diffeomorphisms, where $p$ is the density to the probability space $(\mathcal{X}, \mathscr{B}(\mathcal{X}), \mu)$ that $f$ is applied to. We can now easily evaluate the density of the pushforward measure.

Find good reference to thm

thm: trans

**Theorem 2.3.7.** *Let $\mathcal{Z}_0 \subseteq \mathbb{R}^D$ and $(\mathcal{Z}_0, \mathscr{B}(\mathcal{Z}_0), \mu)$ be the base probability space. Let $f$ be a $C^1_{q_{\boldsymbol{z}_0}}$-diffeomorphism, where $q_{\boldsymbol{z}_0}$ is the density to the base probability space. Then the density of the pushforward distribution induced by $f$, is defined as*

$$q_{\boldsymbol{x}}(\boldsymbol{x}) = \sum_{i=1}^{k} q_{\boldsymbol{z}}(f_i^{-1}(\boldsymbol{x}))\, |det(J_{f_i^{-1}}(\boldsymbol{x}))|, \tag{2.3}$$

{eq: trans}

*where $J_{f_i^{-1}}(\boldsymbol{x})$ is the Jacobian to the function $f_i^{-1}$ evaluated at $\boldsymbol{x}$.*

Hence, we can always evaluate the density to the transformed data, which is one of the major advantages NF has compared to other popular generative models such as GAN. A special case of the Theorem 2.3.7 is when $k = 1$, which gives us the well known formula

$$q_{\boldsymbol{x}}(\boldsymbol{x}) = q_{\boldsymbol{z}}(f^{-1}(\boldsymbol{x}))|\det\left(J_{f^{-1}}(\boldsymbol{x})\right)|.$$

From the fact that the transformation is invertible, allows us also to rewrite the Jacobian above to $[J_f(\boldsymbol{z}_0)]^{-1}$. This follows from the fact that the Jacobian of the identity function $f^{-1}(f(\boldsymbol{z}_0))$ is simply the identity matrix. Applying the chain rule, we have

$$\begin{aligned} I_D = J_{f^{-1}\circ f}(\boldsymbol{z}_0) &= J_{f^{-1}}(f(\boldsymbol{z}_0))\, J_f(\boldsymbol{z}_0) \\ [J_f(\boldsymbol{z}_0)]^{-1} &= J_{f^{-1}}(\boldsymbol{x}), \end{aligned}$$

where the inverse exist as the function is inverse, which means that the determinant of the Jacobian is nonzero, which means the matrix is inverse. This is a minor point, but is rather important when it comes to training. As in a maximum likelihood situation, we wish to send the data backwards towards base density, and we can then calculate the Jacobian of the inverse simultaneously. While in a variational inference situation, we wish to sample data and transform it so we can evaluate the target likelihood. It is then computationally wise to also compute the Jacobian of the forward flow. Hence, the equality can be important in terms of computational speed when implemented.

One strength of flows is to compose several less complex transformations. That is, using $T \in \mathbb{Z}^+$ transformations, compose the flow

$$\begin{aligned} f(\boldsymbol{z}_0) &= f_T \circ f_{T-1} \circ \cdots \circ f_2 \circ f_1(\boldsymbol{z}_0) \\ &= \bigcirc_{t=1}^{T} f_t(\boldsymbol{z}_0). \end{aligned}$$

We refer to $f$ as a flow, $f_t$ for the transformation of the vector $\boldsymbol{z}_{t-1}$, and componentwise tranformation of $z_{t-1,d}$ as $f_{t,d}$ for $d = 1, 2, \ldots, D$. That is, the composition of $T$ transformations is called a *flow*. The benefit of less complex transformations $f_t$ is that estimating the parameters to each one is not as burdensome, and scaling it with more transformations is not necessarily equivalent to approximate many parameters to a complex function. It also allows for sharing of information between the dimensions, while allowing for quick evaluation of density, which we shall come back to in Section 2.4. Using

several transformations, where each transformation may give an easy to calculate Jacobian, means that we can easily find the density of the pushforward measure easily. As applying first the chain rule, we have

$$\det\left(J_{\bigcirc_{t=T}^{1} f_t^{-1}}(\boldsymbol{x})\right) = \det\left(J_{f_1^{-1}}(\boldsymbol{z}_1)\cdots J_{f_{T-1}^{-1}}(\boldsymbol{z}_{T-1})\cdot J_{f_T^{-1}}(\boldsymbol{x})\right),$$

where $\boldsymbol{z}_t = f_{t+1}^{-1}\circ\cdots\circ f_{T-1}^{-1}\circ f_T^{-1}(\boldsymbol{x})$. Using then the fact that for square matrices $A, B$ we have $\det(A\cdot B) = \det(A)\cdot\det(B)$, we get

$$\det\left(J_{\bigcirc_{t=T}^{1} f_t^{-1}}(\boldsymbol{x})\right) = \prod_{t=1}^{T}\det\left(J_{f_t^{-1}}(\boldsymbol{z}_t)\right).$$

In terms of densities, one could have obtained the same result with regards to determinant of compositions, through observing that the input $\boldsymbol{z}_{t-1}$ to $f_t$ also have a density. We can then apply recursively Equation (2.3), and when $k=1$ for all transformations, we have the base density times $\prod_{t=1}^{T}\det\left(J_{f_t^{-1}}(\boldsymbol{z}_t)\right)$.

We can now define normalizing flows formally for the purposes of this thesis. This will not be all-encompassing, as we are working with a specific probability space and with discret time-steps in our flow, i.e $t\in\mathbb{Z}^+$. There are other flows defined for continuous time and with discret distributions etc. However, for the moment this definition will cover the majority of flows and usecases in the literature.

We may touch upon it later if time allows us, e.g continuous time

**Definition 2.3.8.** Let $\mathcal{A} = (\mathcal{Z}_0, \mathscr{B}(\mathcal{Z}_0), \mu)$ be a probability space with $\mathcal{Z}_0\in\mathbb{R}^D$. Let $f_t$ be a piecewise $C^1$-diffeomorphism w.r.t $q_{\boldsymbol{z}_{t-1}}$ for all $t=1,2,\ldots,T$. A *normalizing flow* (NF) is defined by $(\mathcal{A}, f)$, where $\mathcal{A}$ is the base probability space and $f=\bigcirc_{t=1}^{T} f_t$ is the flow. The induced density by letting a sample $\boldsymbol{z}_0$ from $\mathcal{A}$ flow through $f$ is then given by

$$q_{\boldsymbol{z}_T}(\boldsymbol{z}_T) = \sum_{i_T=1}^{k_T}\cdots\sum_{i_1=1}^{k_1} q_{\boldsymbol{z}_0}\left(\bigcirc_{t=T}^{1} f_{t,i_t}^{-1}(\boldsymbol{z}_T)\right)\prod_{t=1}^{T}\det(J_{f_{t,i_t}^{-1}}(\boldsymbol{z}_t)), \tag{2.4}$$

where $f_{t,i_t}$ is the diffeomorphism of tranformation $t$ over partition $i_t$.

When $f$ is a $C^1$-diffeomorphism, we get the induced density

$$q_{\boldsymbol{z}_T}(\boldsymbol{z}_T) = q_{\boldsymbol{z}_0}\left(\bigcirc_{t=T}^{1} f_t^{-1}(\boldsymbol{z}_T)\right)\prod_{t=1}^{T}\det(J_{f_t^{-1}}(\boldsymbol{z}_t)).$$

Notice that in the definition of NF we have not included anything regarding the target distribution. Eventhough we often speak about a flow and a target distribution, the flow is simply defined by transforming samples from a base distribution in such a manner that we can also evaluate the induced density. The application of flows will necessarily be concerned with target distribution and minimazation of some sort of measurement between target and flow induced density. One can also ask questions about a particular flow and its capabilities/flexibilities w.r.t a target distribution. But ultimately, any combination of $(\mathcal{A}, f)$ defined as above is a flow, no matter how trivial or impractical the resulting distribution is.

We know wish to find transformations $f$ by considering the following points.

- Flexible and expressive, such that we can always transfrom from $\mathcal{A}$ to any distribution as described above.
- Limit number of parameters to estimate.
- Computation wise, cheap to compute both inverse and determinants.

Obviously there may be some compromises between the first and the other two points. Our goal is then to construct flows such that one allow for high expressitivity while remain computationally feasible. of the flows.

### Analytical, Tractable and Intractable Inverse

Ought we define a bit better what these terms mean?

## 2.4 Flow Structure

seq: struct

When constructing transformations $f$ one has to choose the form of the function $f_{t,d}$, as well as the structure. By structure we mean what variables $z_{i,j}$ is needed to calculate the transformation. This leads to the unfortunate situation that one speaks about a particular flow, while intending only to speak of the structure. We decouple these two, and introduce the form of the function in the next section. Some of the most popular NF's can use a myriad of structures, which we shall formalizer here. So while some flows in the literature only allow for a specific structure, others allow for a larger class of them.

We introduce, for ease of readibility, $\mathcal{T} = \{1, 2, \dots, T\}$ and $\mathcal{D} = \{1, 2, \dots, D\}$.

def: struct

**Definition 2.4.1.** Let $(\mathcal{A}, f)$ be a normalizing flow. A *flow-structure* is defined as a mapping

$$\mathcal{S} \colon \mathcal{T} \times \mathcal{D} \to \mathcal{P}(\mathcal{T} \times \mathcal{D}),$$

such that $\mathcal{S}(t, d)$ is the set indicating which variables in the flow that are used to calculate $z_{t,d}$.

*Remark* 2.4.2. Obviously, the set $\mathcal{S}(t, d)$ for any $(t, d)$ is never empty, as it is always dependent on $(t-1, d)$. There are cases where it is useful to talk about the flow-structure without the element $(t-1, d)$, in which we refer to the mapping as $\mathcal{S}_{ext}$. That is,

$$\mathcal{S}_{ext}(t, d) = \mathcal{S}(t, d) \setminus \{(t-1, d)\}.$$

Elements $(i, j) \in \mathcal{S}_{ext}(t, d)$ is such that $f_{t,d}(z_{t-1,d})$ also uses $z_{i,j}$. The way it is included in the transformation $f_{t,d}$ can vary a lot, as we shall see in the next section. However, thinking about the definition and goals of NF, one quickly finds out that there are limitations on $\mathcal{S}$. Both in terms of invertibility, but also computationally w.r.t determinant etc. Before we explore this any further, we introduce a different way to view $\mathcal{S}$; as a graph.

Let $(\mathcal{A}, f)$ be a normalizing flow with an accompanied flow-structure $\mathcal{S}$. We define a graph $G$ with verticies $V = \mathcal{T} \times \mathcal{D}$ and directed edges $E$ with an edge $(t, d)$ from all verticies in $\mathcal{S}(t, d)$. It is sometimes useful to separate edges from

$(t-1,d)$ to $(t,d)$ from the rest, here as well. We shall therefore refer to edges on the former form to be in $E_{int}$ and the rest in $E_{ext}$, with $E = E_{int} \cup E_{ext}$[1]. We abuse the notation somewhat, and allow $\mathcal{S}$ to both be referred to as the mapping in Definition 2.4.1 and also the graph it induces, with the context deciding which one. Typically, if we speak about $\mathcal{S}$ itself, we tend to do it through graph $G$. When we are talking about a specific variable and what it is dependent on, we refer to the map $\mathcal{S}$ and the set it outputs for a given variable.

### DAGs

write about why the flow-structure must be a DAG

### Triangular

define generally triangluar structures and prove triangular jacobian

Two important examples of triangular flow-structures, of which are often used in the literature (Papamakarios et al. 2017, Kingma et al. 2016, Huang et al. 2018), are based off autoregressive models.

**Definition 2.4.3.** Let $(\mathcal{A}, f)$ be a NF with accompanying flow-structure $\mathcal{S}$. If there exist a permutation $\phi_t \colon \mathcal{D} \to \mathcal{D}$ for all $t \in \mathcal{T}$ such that

$$\mathcal{S}_{ext}(t, \phi_t(d)) = \{(t,i) \mid i \in \mathcal{D} \text{ and } \phi_t(i) < \phi_t(d)\}$$

then $\mathcal{S}$ is an *autoregressive flow-structure* (AR flow-structure).

The autoregressive flow-structure simply says that the transformation of $z_{t,d}$ is all based on all the variables already transformed and $z_{t-1,d}$, where the ordering of the dimensions is decided by the permutation $\phi_t$. If, for all transformations, we have that the $\phi_t$ is the idendity permutation, then we call it AR-structure without permutation. Including permutation will in some cases be crucial to allow for flexible transformations in all dimensions, and can be seen as a form of information sharing between the dimensions. We also include the inverse AR-structure, which can be interpreted as AR-structure when we do inverse flow.

**Definition 2.4.4.** Let $(\mathcal{A}, f)$ be a NF with accompanying flow-structure $\mathcal{S}$. If there exist a permutation $\phi_t \colon \mathcal{D} \to \mathcal{D}$ for all $t \in \mathcal{T}$ such that

$$\mathcal{S}_{ext}(t, \phi_t(d)) = \{(t-1,i) \mid i \in \mathcal{D} \text{ and } \phi_t(i) < \phi_t(d)\},$$

then $\mathcal{S}$ is an *inverse autoregressive flow-structure* (IAR flow-structure).

## 2.5 Transformations

introduce transformations

[1] *int* is short for interior, and refers to the fact that $z_{t,d}$ is directly transformed from the interior variable. While other variables gives "help" with the transformation from the "outside", hence exterior (*ext*).

### Affine

One of the first transformations that was introduced was a simple affine transformation, componentwise. It was first introduced using an IAR flow-structure and later also with an AR flow-structure (Kingma et al. 2016; Papamakarios et al. 2017). One of the reasons they have been popular with the aformentioned structures are partly due to the fact that the resulting flow can be inverted easily. However, the form of the function can easily be applied to many flow-structures. Which is why we define it more generally here.

**Definition 2.5.1.** Let $(\mathcal{A}, f)$ be a normalizing flow, with accompanied flow-structure $\mathcal{S}$. An *affine transformation* has the form

$$z_{t,d} = f_{t,d}(z_{t-1,d}) = a_{t,d}\, z_{t-1,d} + b_{t,d},$$

with $a_{t,d}, b_{t,d} = \mathcal{H}(\mathcal{S}_{ext}(t,d))$. We refer to $(\mathcal{A}, f)$ as an *affine flow*, if $f_{t,d}$ is an affine transformation for every $t \in \mathcal{T}$ and $d \in \mathcal{D}$.

Here $\mathcal{H}$ can be any function with output $a, b$ and input based off the exterior structure to the flow. We typically leverage the universality off neural networks and let $\mathcal{H} = \mathcal{NN}$, for a certain number of hidden layers, activiation functions, and number of neurons.

write more about affine and what properties of $\mathcal{S}$ gives analytical inverse.

### Neural Network Transformation

Transforming the variables by using a neural network is not trivial, as one must be able to guarantee invertibility. One way to assure this was done by Huang et al. 2018, where they constrain the neural network that transform $z_{t,d}$ to a netowork with bijective activiation functions and nonnegative weights. We denote these networks here by $\mathcal{NN}^+$.

**Definition 2.5.2.** Let $(\mathcal{A}, f)$ be a normalizing flow, with accompanying structure $\mathcal{S}$. A $\mathcal{NN}^+$*-transformation* has the form

$$z_{t,d} = f_{t,d}(z_{t-1,d}) = \mathcal{NN}^+(z_{t-1,d}),$$

with the weights of the network defined by $W = \mathcal{H}(\mathcal{S}_{ext}(t,d))$. A normalizing flow is a $\mathcal{NN}^+$*-flow* if for all $t \in \mathcal{T}$ and $d \in \mathcal{D}$, the transformation $f_{t,d}$ is a $\mathcal{NN}^+$-transformation.

## 2.6 Universality

An aspect of normalizing flows is to know whether or not we can approximate any distribution by simple increasing some $\phi$, where $\phi$ can be either one or several parameters determined before training. In some cases such as a $\mathcal{NN}^+$-flow, we may talk about the size of the neural network. With splines it may be the number of knots used. Another apt parameter in the context of normalizing flows is the number of transformations. The important part is that we can guarantee convergence in distribution given flow, by increasing some parameters $\phi$. We also denote $\boldsymbol{z}_T^{(\phi)}$ as the result of a corresponding flow $f$ using $\phi$-neurons/knots/transformation etc. It is worth noting that even with this guarantee, we may not do well with any given target distribution in practice, as

we have finite data and computing power to estimate the parameters. This is quite analogous to neural networks in general, as we know we can approximate any continuous function when the input is compact. However, when working with images one introduces an inductive bias regarding locality which results in a Convolutional Neural Network (CNN). Regardless, knowing that a given flow is universal w.r.t distribution tells us that the target space induced by the flow is not restricitive. We follow Huang et al. 2018 when formally defining universality.

**Definition 2.6.1.** Let $(\mathcal{A}, f)$ be a normalizing flow, $p_{\boldsymbol{x}}$ be any target density in a class of densities $\mathscr{P}$, and $\phi$ be a set of parameters which by increasing also increases the size of the model. The flow $f$ is an *universal density approximator* (UDA) for $\mathscr{P}$ if there exist a sequence of $\boldsymbol{z}_T^{(\phi)}$, such that when $\phi \to \infty$, $\boldsymbol{z}_T^{(\phi)} \xrightarrow{d} \boldsymbol{x}$.

A couple of remarks are in order. Firstly, This definition is quite analogous to how we speak about universal approximators with regards to neural networks. We have a certain network architecture and class of functions, namely continuous functions. We then posit existence of a specific network which converges to any function in the class uniformly, when increasing the model through increasing $\phi$, where $\phi$ can in this case be number of layers or number of neurons. Secondly, we do recognize that the name *universal density approximator* is a little bit misleading. If something converges in distribution does not imply convergence in density. This can be illustrated with a density

$$p_n(\boldsymbol{x}) = \begin{cases} 1 - \cos(2\pi n x), & \text{if } 0 < x < 1 \\ 0, & \text{elsewhere} \end{cases}$$

This does clearly not converge to any density, but the CDF of $p_n$ is equal to $x - \frac{\sin(2\pi n x)}{2\pi n}$ which converges to $x$ when $n \to \infty$. Hence, it converges to the Uniform distribution between 0 and 1, while the density does not converge to 1. One might ask how well the density we achieve from the flow corresponds to the target density, even in flows that are UDA. Through the way one train the flow, and its use of KL-divergence, makes it less probable that we converge in distribution while not being reasonably close density wise. It may, however, affect the ability of finding the parameters such that we have convergence in distribution. We therefore ought to be cautious of the property UDA and what is conveys, while still appreciate the guarantee that we can in theory converge to any continuous target distribution. Also worth noting that if one is able to show that the density of a flow can converge to any target density, then the flow is an UDA. This follows from Scheffé's Theorem, which states that convergence in density a.e implies convergence in distribution (Scheffe 1947). Regardless, we follow the literature and its definition of an UDA, while keeping in mind the gap between convergence of density and in distribution.

### NAF

As alluded to earlier, $\mathcal{NN}^+$-flows are UDA. To be more specific, a certain model in the class of flows referred to as NAF is an UDA for $\mathscr{P}$, where $\mathscr{P}$ is the class of positive, continuous densities. To quickly summarize the result presented by

Huang et.al: The flow given by an transformation function

$$f_d(\boldsymbol{z}_0) = \sigma^{-1}\left(\sum_{j=1}^{n} w_{d,j}\sigma\left(\frac{z_{0,d} - b_{d,j}}{\tau_{d,j}}\right)\right), \text{ for all } d \in \mathcal{D} \tag{2.5}$$

{eq:naf}

where the parameters $(w_{d,j}, b_{d,j}, \tau_{d,j})_{j=1}^{n}$ is given by $\mathcal{H}(\mathcal{S}_{ext}(t,d))$. The flow-structure is an IAR-structure and $\mathcal{H}$ is a deep neural network. The flow described is referred to as NAF-DSF. It is only one transformation, and all its expressiveness rely on increasing number of neurons in the hidden layer, $n$. With this flow, Huang et.al proved the following theorem.

thm: nafUDA

**Theorem 2.6.2** (Huang et al. 2018)**.** *Let $\boldsymbol{z}_0$ be a random vector in an open set $\mathcal{U} \subset \mathbb{R}^D$. Let $\boldsymbol{x}$ be a random vector in $\mathbb{R}^D$ with its density being a member of*

$$\mathscr{P} = \{p_{\boldsymbol{x}} \mid p_{\boldsymbol{x}} \in C(\mathbb{R}^D; \mathbb{R}_*^+)\}.$$

*Then there exist a transformation of $\boldsymbol{z}_0$ to $\boldsymbol{z}_1$ through NAF-DSF such that $\boldsymbol{z}_1 \xrightarrow{d} \boldsymbol{x}$.*

Hence, NAF-DSF is an UDA. However, as mentioned earlier, the inverse of NAF is only tractable. A natural question is then whether or not a flow with analytical inverse can be an UDA.

### Affine transformation

When it comes to flows with affine transformation, is has for a long time been an open question whether or not the flow is an UDA. This year Wehenkel and Louppe gave a counter-example in terms of a target density that cannot be approximated by affine flow. The goal of this section is to both refute the counter-example for a general $D$, and also prove that affine flow is not an UDA for $D = 1$. We start by refreshing what an affine normalizing flow is.

$$z_{0,d} \sim q_{\boldsymbol{z}_0} \tag{2.6}$$

$$z_{t,d} = a_{t,d} \cdot z_{t-1,d} + b_{t,d} \tag{2.7}$$

{eq: affine_ trans}

for all $t \in \mathcal{T}$ and $d \in \mathcal{D}$, and with $(a_{t,d}, b_{t,d}) = \mathcal{H}(\mathcal{S}_{ext}(t,d))$. With the the following constraints, that all $a$'s are positive, and that the structure is with permutations, i.e all dimensions have more than one ancestor in the corresponding graph. If this is not the case, then there exist a $d$ such that $z_d$ is an affine transformation of constants $a$'s and $b$'s, which is detrimental to the flows expressiveness, as we shall see.

A paper emerged (Wehenkel et al. 2020), which claimed that affine flow cannot be an UDA, regardless of structure or number of transformations. The proof is by a counter-example, which state a target density where one component is independent, and with base density being a Gaussian with diagonal covariance matrix. That is, a target distribution with $\boldsymbol{x} \sim p_{\boldsymbol{x}}$ such that $x_j \perp\!\!\!\perp x_{-j}$, for at least one $1 \leq j \leq D$. The first argument that if the flow is construced such that $z_j$ only have one ancestor, it cannot be an UDA. This is correct, as then $z_j$ becomes Gaussian. However, with a reasonable structure that makes $z_j$ have more than itself as ancestor, it is certainly not Gaussian (not even conditionally). The claim when the component has more than one ancestor is that it either

must hurt the bijectivity of the flow, or independence. When the expectation and variance for $p_{\boldsymbol{x}}$ exist, we can show that this is not the case. We then claim that this also holds for when expectation and variance does not exist.

The idea for the proof is to construct part of the flow, and show that such flow exist. Then we are assuming that we can always find an affine flow from any non-independent distribution to another. Then it follows that any independent components in either base distribution, target distribution or both is not a problem given that affine flow is an UDA for non-independent components. We have to diverge from the base distribution of Gaussian. This is simply to ensure we are sampling from a compact set, i.e closed and bounded on $\mathbb{R}^D$. However, starting for example with a standard multivariate Gaussian can also be approximated very well with a set that is large, yet bounded. Hence, theoretically compactness is crucial, as it allow us to use universal approximation w.r.t deep neural networks. Yet practically, it ought not be too much difference from a very large compact set and sampling from a standard Gaussian.

We are constructing part of the flow, which is the two first and two last parts of the flow. We are working with a target distribution where each component is independt of the others and equally with a base distribution. Other versions with different causality follows directly from this case. The flow we are working with start with the first transformation

$$z_{1,d} = a(z_{0,1:d-1})\, z_{0,d} + b(z_{0,1:d-1}) = z_{0,d} - \mu_{z_{0,d}}$$

where $\mu_{z_{0,d}}$ is the mean of the base density for the $d$th dimension. It then follows with

$$z_{2,d} = a(z_{1,1:d-1})\, z_{1,d} + b(z_{1,1:d-1}) = z_{1,d} + \sum_{j=1}^{d-1} z_{1,j}.$$

The final transformations are

$$z_{T-1,d} = a(z_{T-1,1:d-1})\, z_{T-2,d} + b(z_{T-1,1:d-1}) = z_{T-2,d} - \sum_{j=1}^{d-1} z_{T-1,j},$$

and finally

$$z_{T,d} = a(z_{T,1:d-1})\, z_{T,d} + b(z_{T,1:d-1}) = z_{T,d} + \mu_{z_{T,d}}.$$

Notice that the structures for the first two steps are inverse autoregressive, while the latter two uses autoregressive structures. However, the last transformations can be written with an inverse autoregressive structure by simply doing $2 * D$ transformations instead of 2, where for each transformation $t$, one variable $d$ is transformed by as defined above, and the other dimensions are simply transformed by the identity function. Preferring fewer transformation for readability, we stick with the transformation defined above, but noting that we can have an inverse autoregressive structure. Additionally, we are going to talk about the density of the inverse transformation, that is $q^{-1}_{\boldsymbol{z}_{T-2}}$. That is, the density induced density of

$$z_{T-2,d} = (z_{T,d} - \mu_{z_{T,d}}) + \sum_{j=1}^{d-1} (z_{T,j} - \mu_{z_{T,j}}).$$

We assume now that $\boldsymbol{z}_D \sim p_{\boldsymbol{x}}$. Obviously in reality, we would not know what the target distribution $p_{\boldsymbol{x}}$ is, however we are here only interested to show that there exist an affine flow from $p_{\boldsymbol{x}}$ to some non-independent distribution. As the inverse is also an affine flow, we can then use the existence of the inverse flow to construct a flow from non-independent to independent distribution. By then assuming that the affine flow is an UDA between non-independent distributions, we can show existence of flow between independent distributions through our construction.

Let us then consider the density that the flow from transformation 0 to 1 induces, and the density induced by the inverse flow from $T$ to $T-1$. Since we are working with independence in both base and target density, we have the induced densities

$$q_{\boldsymbol{z}_1}(\boldsymbol{z}_1) = \prod_{d=1}^{D} q_{z_{0,d}}(z_{0,d})$$

and

$$q^{-1}_{\boldsymbol{z}_{T-2}}(\boldsymbol{z}_{T-2}) = \prod_{d=1}^{D} q_{z_{T,d}}(z_{T,d}) = \prod_{d=1}^{D} p_{x_d}(x_d)$$

where the last equality stems from the assumption above. Hence the density have not changed. However, we have added dependence.

lemma: corr_densi

**Lemma 2.6.3.** *Let the expectation and variance exist for both the base and target density. The two densities $q_{\boldsymbol{z}_2}$ and $q^{-1}_{\boldsymbol{z}_{T-2}}$ are both not independent.*

*Proof.* We only prove for one of the transformation, as the proof is identical. The expectation of any variable in $\boldsymbol{z}_1$ is zero. That is,

$$E(z_{2,d}) = E\left(z_{1,d} + \sum_{j=1}^{D} z_{1,j}\right) = E\left(z_{0,d} - \mu_{z_{0,d}} + \sum_{j=1}^{D} z_{0,j} - \mu_{z_{0,j}}\right) = 0$$

Which means that the covariance between any two variables $z_{2,i}$ and $z_{2,j}$, where w.l.g $i < j$, is

$$\begin{aligned}\mathrm{Cov}(z_{2,i}, z_{2,j}) &= E\left(\left(z_{1,i} + \sum_{k=1}^{i-1} z_{1,j}\right)\left(z_{1,j} + \sum_{l=1}^{j-1} z_{1,l}\right)\right) \\ &= E\left(\left(\sum_{k=1}^{i}\sum_{l=1}^{i} z_{1,k}\, z_{1,l}\right) + \left(\sum_{k=1}^{i}\sum_{l=i+1}^{j} z_{1,k}\, z_{1,l}\right)\right).\end{aligned}$$

The expectation of the last sum is 0, as $\boldsymbol{z}_1$ are independent (only subtracted mean from each dimension). The first sum also end up in 0 when $k \neq l$, hence

$$\begin{aligned}\mathrm{Cov}(z_{2,i}, z_{2,j}) &= E\left(\sum_{k=1}^{i} z_{1,k}^2\right) \\ &= E\left(\sum_{k=1}^{i} (z_{0,k} - \mu_{z_{0,k}})^2\right) = \sum_{k=1}^{i} \mathrm{Var}(z_{0,k}) > 0.\end{aligned}$$

Which means that the induced distributions with density $q_{\boldsymbol{z}_2}$ and $q_{\boldsymbol{z}^{-1}_{T-2}}$ are not independent. ■

The next step before proving that such flows described above exist, given that we have affine flow being UDA for non-independent distributions, is to make sure we have compactness after each flow. This is because we are using a deep neural network to estimate the $a$'s and $b$'s for each transformation.

lemma: compact

**Lemma 2.6.4.** *Let the set of values where the base density is greater than 0 be compact in $\mathbb{R}^D$. Then the output is compact in $\mathbb{R}^D$ for every transformation $t = 1, 2, \cdots, T$ on an affine flow with inverse autoregressive structure.*

*Proof.* A set in $\mathbb{R}^D$ is compact if and only if it is bounded and closed. Assuming $\boldsymbol{z}_{t-1} \in \mathcal{Z}_{t-1}$, where $\mathcal{Z}_{t-1}$ is a compact set in $\mathbb{R}^D$. Using, w.l.g, an inverse autoregressive structure where $z_{t-1,d}$ is dependent on $z_{t-1,1:d-1}$.

To show preservation of compactness, we simply need to show that the transformation in Equation (2.7) is continuous. Transformation of the first dimension is clearly continuous, as the parameters $a_{t,1}, b_{t,1}$ is is constant for all $z_{t-1,1}$. Next, assuming the structure uses activation function ReLU, and final activation function on $a$ to make it positive is continuous, we have that the neural network with input $z_{d-1,1:t-1}$ is continuous. As it simply is a composition of affine transformation with added ReLU (which is continuous), and an extra continuous activation function on $a$. It therefore follows that the transformation of $z_{t-1,d}$ is also continuous. Hence, the transformation of $\boldsymbol{z}_{t-1}$ is continuous. Any continuous function from a compact subset of a metric space to another metric space implies the image of the function is also compact. ■

This allows us to always be able to find the transformations for $t = 1, 2, T-2, T-1$, using inverse autoregressive structures and allowing for the neural network associated with each structure to be as large as one needs. This is due to the fact that we know input is compact and output is either constant, e.g subtracting mean, or sum of the input, which are both continuous. Since the flows we wish to approximate are all continuous, hence we know we can find them using neural networks by Theorem 1.0.2.

**Proposition 2.6.5.** *Let $\mathcal{Z}_0 \in \mathbb{R}^D$ be compact. If affine flow with inverse autoregressive structure (with permutations) is an UDA when both base and target distribution are non-independent, then it is an UDA for all base and target densities with finite expectation and variance.*

*Proof.* As noted earlier, we are only proving it for when both base and target density are independent, since it follows from this that it holds when either one is non-independent distributions. The flow goes as follows. Sample from $\mathcal{Z}_0$, as the input is compact and both $a$ and $b$ in the first flow are constants, means that Theorem 1.0.2 can always approximate the first transformation

$$z_{1,d} = a(z_{0,1:d-1})\, z_{0,d} + b(z_{0,1:d-1}) = z_{0,d} - \mu_{z_{0,d}}.$$

From Lemma 2.6.4 and the fact that sum of input from a subset in $\mathbb{R}^D$ is an continuous function, which means we can approximate transformation number two arbitraily well by Theorem 1.0.2. That is, we can approximate arbitraily well

$$z_{2,d} = a(z_{1,1:d-1})\, z_{1,d} + b(z_{1,1:d-1}) = z_{1,d} + \sum_{j=1}^{d-1} z_{1,j}.$$

Through Lemma 2.6.3 and assumption of UDA between non-independent distributions, we have that we can find a flow such that the sample $\boldsymbol{z}_{T-2}$ converges in distribution to the distribution with density $q^{-1}_{\boldsymbol{z}_{T-2}}$. Through the same reasoning as with the first two transformations, assuming we rewrite the two last transformations such that it have an inverse autoregressive structure. As we know $\mathcal{Z}_{T-2}$ is compact in $\mathbb{R}^D$ due to Lemma 2.6.3, which means that we can arbitraily well approximate the transformations

$$z_{T-1,d} = a(z_{T-1,1:d-1})\, z_{T-2,d} + b(z_{T-1,1:d-1}) = z_{T-2,d} - \sum_{j=1}^{d-1} z_{T-1,j}$$
$$z_{T,d} = a(z_{T,1:d-1})\, z_{T,d} + b(z_{T,1:d-1}) = z_{T,d} + \mu_{z_{T,d}}.$$

By definition of the induced $q^{-1}_{\boldsymbol{z}_{T-2}}$, i.e it is the inverse transformation is such that $\boldsymbol{z}_T \sim p_{\boldsymbol{x}}$. Which means that $\boldsymbol{z}_0 \xrightarrow{d} \boldsymbol{x}$, which means that the affine flow is also an UDA for base and target distributions that are independent with expectation and variance existing. ■

This means that, at least for the case when we have existence of expectation and variance, the counter-example given by Wehenkel et al. 2020 does not necessarily hold. We add necessarily, as we are currently working with base distribution defined on a compact set, which we allow as a fair condition to put on base densities. Although we do acknowledge that the counter-example was based off independent Gaussian, but rebuking our proposition due to the assumption of compactness is to refute all proofs of UDA in the literature. As all proofs assume this, due to the limitations in Theorem 1.0.2.

conj: nonind

**Conjecture 2.6.6.** *There always exist an affine flow with inverse autoregressive flow-structure, such that any independent continuous distribution can be transformed through the flow, where the induced distribution is non-independent.*

If this hold, then we strengthen our proposition such that if affine flow with inverse autoregressive structure is an UDA between non-independent distributions, then it is also UDA for all distributions given compact $\mathcal{Z}_0$. As we do not know Conjecture 2.6.6, means that Wehenkel et al. 2020 cannot be known as well.

However, we can easily prove that for $D = 1$, the affine flow cannot be an UDA.

**Proposition 2.6.7.** *Let the target density be $p_{\boldsymbol{x}}$ with $D = 1$. Then an affine flow cannot be an UDA, regardless of flow-structure.*

*Proof.* Due to the fact that all the parameters $\{a_{t,1}, b_{t,1}\}_{t=1}^{T}$ in the flow are constants, as the set $\mathcal{E}_{ext}$ of the structure is empty. This means that the induced density of the flow is

$$q_{\boldsymbol{z}_T} = q_{\boldsymbol{z}_0}(f^{-1}(\boldsymbol{z}_T)) \prod_{t=1}^{T} |a_{t,1}|^{-1}.$$

This means for instance that the number of modes that we originally have in $q_{\boldsymbol{z}_0}$ is preserved, as we are multiplying by the same constant for all $f^{-1}(\boldsymbol{z}_T)$. This means we need to know a priori the number of modes the target distribution

have so we can do the same for our base distribution. However, this breaks with the assumptions behind the base distribution. Hence, the flow $f$ cannot be an UDA. ■

It still remains an open question whether or not the affine flow is an UDA for $D > 1$.

### Analytical Inverse Flows are UDA

Affine flow with FILLER flow-structure is just one of a whole class of non-linear analytically invertible normalizing flows. This class can be written as

$$f_{t,d}(\boldsymbol{z}_{t-1}) = c_{t,d} \cdot h_{t,d}[a_{t,d} \cdot z_{t,d} + b_{t,d}] + d_{t,d}, \tag{2.8}$$

{eq: aif_flow}

where $h_{t,d}$ are piecewise $C^1$-diffeomorphisms, analytically invertible, and

$$(a_{t,d}, b_{t,d}, c_{t,d}, d_{t,d}) = \mathcal{H}(\mathcal{S}_{ext}(z_{t,d})).$$

It is easy to see from this definition that affine flow is just one of many possible flows that follows Equation (2.8). We shall refer to the flows that uses transformations on the form Equation (2.8) as AIFs.

Write about benefits of introducing h

We now turn our attention to one certain AIF, which we will then show is an UDA. As far as we know, this have never been done. That is, to show that analytical inverse flows can be UDAs.

**Definition 2.6.8.** Let $(\mathcal{A}, f)$ be a normalizing flow with corresponding flow-structure $\mathcal{S}$. Let $q_{\boldsymbol{z}_0}$ be the density that correpsponds to the base distribution $\mathcal{A}$. A piecewise affine flow (PAF), is defined as

$$\boldsymbol{z}_0 \sim q_{\boldsymbol{z}_0} \tag{2.9}$$
$$f_{0,d}(z_{0,d}) = \sigma(z_{0,d}) \tag{2.10}$$
$$f_{t,d}(\boldsymbol{z}_{t-1,d}) = h_{t,d}(z_{t-1,d} - b_{t,d}) + b_{t,d} \tag{2.11}$$
$$f_{T+1,d}(z_{T,d}) = \hat{\sigma}(z_{T-1,d}) \tag{2.12}$$
$$f_{T+2,d}(z_{T+1,d}) = \sigma^{-1}(z_{T,d}). \tag{2.13}$$

{eq: h_trans}

where

$$h_{t,d}(x) = \begin{cases} a_{t,d} \cdot x, & \text{if } x > 0 \\ x, & \text{else,} \end{cases} \tag{2.14}$$

and

$$\hat{\sigma}(x) = 2\left(\sigma(x) - \frac{1}{2}\right). \tag{2.15}$$

The parameters to estimate in the flow is simply $(a_{t,d}, b_{t,d}) = \mathcal{H}(\mathcal{S}_{ext}(z_{t,d}))$, where $a_{t,d} > 0$, for $t \in \mathcal{T}$ and $d \in \mathcal{D}$.

The indexing of steps $t$ is done such that $T$ decides how many transformations of the form Equation (2.11). The three other transformations are always there and does not depend on whether one adds more or fewer transformations, and

TODO: Change flow-structure FILLER word when it is defined properly

they do not depend on any other variables than the one it transforms. Which means we relate T the the type of transformations that one can add more of, which simply allow us to avoid always refering to $T-3$ transformations etc. in the following proofs.

We now give an explanation of the flow. It first samples from our base distribution, then uses the standard logistic function $\frac{1}{1+\exp(-x)}$ to map our samples to $(0,1)$. Note that the transformation can be any as long as the output is positive (we chose somewhat arbitraily the sigmoid function). We then apply $T$ transformations of Equation (2.11). The transformation can be seen as a reverse leaky ReLU, but where the point that used to be simply 0, is decided by $b_{t,d}$. We then, as $z_{T,d} > 0$ for all $d \in \mathcal{D}$, map the samples $z_{T,d}$ from $(0,\infty)$ to $(0,1)$ through $\hat{\sigma}$, before applying the standard logit function, i.e inverse to the standard logistic function.

**Proposition 2.6.9.** *Let $(\mathcal{A}, f)$ be a normalizing flow with flow-structure $\mathcal{S}$. If the flow is an PAF and the structure is a FILLER, then the resulting NF is an analytical inverse normalizing flow.*

*Proof.* All transformations apart from Equation (2.11) is well known diffeomorphisms and analytical invertible, no matter the flow-structure as the $\mathcal{S}_{ext}$ is empty for all dimensions given $t \in \{0, T+1, T+2\}$. Hence, we only need to show that the transformation defined in Equation (2.11) as piecewise $C^1$-diffeomorphisms and that the inverse can be written in close form. Let $t \in \{2, \dots, T-1\}$, $f_{t,d}(z_{t-1,d}) = h_{t,d}(z_{t-1,d} - b_{t,d}) + b_{t,d}$ and assume we know $z_{t,d}$ and all variables corresponding to $\mathcal{S}_{ext}(z_{t,d})$. As the structure is a FILLER, means we can guarantee this as long as the transformation itself is invertible. We now need to show, given this, that we can obtain $z_{t-1,d}$, and as a closed form expression. As

$$\lim_{z_{t,d} \to b_{t,d}^-} f_{t,d} = \lim_{z_{t,d} \to b_{t,d}^+} f_{t,d},$$

means it is continuous. As the derivative when constrained to $z_{t-1,d} < b_{t,d}$ is equal to 1, $z_{t-1,d} > b_{t,d}$ is equal to $a_{t,d}$ and $a_{t,d} > 0$ means it is a piecewise $C^1$-diffeomorphism.

The inverse can easily be written in closed form due to the fact that $z_{t-1,d} > b_{t,d} \iff z_{t,d} > b_{t,d}$. As we can obtain $(a_{t,d}, b_{t,d})$ without knowing $z_{t-1,d}$, due to the structure being FILLER. Combining these two means that, given $z_{t,d}$, one can obtain $b_{t,d}$, which gives us $z_{t,d} - b_{t,d}$. Being larger than 0 or not leads us to either divide or not by $a_{t,d}$, which we have obtained without $z_{t-1,d}$. Hence, the inverse can be written as

$$f_{t,d}^{-1}(z_{t,d}) = \begin{cases} \frac{z_{t,d}-b_{t,d}}{a_{t,d}} + b_{t,d}, & \text{if } z_{t,d} - b_{t,d} > 0 \\ z_{t,d}, & \text{otherwise.} \end{cases}$$

Which shows that the transformation is analytically invertible. Hence, the NF with flow being PAF and a structure that is FILLER, is a flow in AIF. ■

## Proof of UDA

We are now going to prove that a normalizing flow with PAF flow and FILLER flow-structure is in fact UDA. The flow-structure we are limiting ourselves to is

an IAR-structure without permutations, and later on prove it for a larger class of structures. This also allows us to talk about $\mathcal{H}_d(z_{0,1:d-1}$, and not a function for each transformation $t$, as they are deterministically defined given $\boldsymbol{z}_0$.

As we are interested in the expressiveness when the number of transformations increase, it makes sense to introduce $g_T$. This refers to the PAF flow without the last logit transformation. Also, the $T$ denotes here the number of transformations of Equation (2.11), and not the $T$th transformation, as we are used to with $f$. We also refer to $\mathcal{H}$ as the function that output the parameters in $g_T$. We are also limiting $\mathcal{A}$, in this section, to have density with the property

$$\begin{cases} q_{\boldsymbol{z}_0}(\boldsymbol{z}_0) > 0, & \text{if } \boldsymbol{z}_0 \in [k_0, k_1]^D \\ q_{\boldsymbol{z}_0}(\boldsymbol{z}_0) = 0, & \text{otherwise,} \end{cases}$$

where $k_0 < k_1$ and $k_0, k_1 \in \mathbb{R}$.

We start by proving that $g_T$ can, for one dimension, find a flow that converges to any monotonically increasing function from a compact set $[k_0, k_1]$ to $[l_0, l_1]$, where the endpoints $k_0, k_1$ maps to $l_0, l_1$ respectively and the functions image is between 0 and 1 including. This is the first step to show UDA, which will rely on approximating the conditional cumulative distribution, before applying the logit function in the last step of the PAF flow.

lemma: paf_one_ dim

**Lemma 2.6.10.** *Let $(\mathcal{A}, g_T)$ be a PAF without the $T+2$ transformation, with an IAR-structure and dimension $D = 1$. Let $g\colon [k_0, k_1] \to [l_0, l_1]$ be a monotonically increasing function with $g(k_0) = l_0$ and $g(k_1) = l_1$, and $0 \leq l_0 < l_1 \leq 1$ . Then there exist a flow on the form $g_T$ that converges uniformly to $g$, when $T \to \infty$. The flow can also be approximated arbitrarily well by a neural network with 4 neurons in each hidden layer.*

*Proof.* For any $\epsilon > 0$, we set $M = \lceil \frac{1}{\epsilon} \rceil$. Start by dividing $[0, 1]$ into $M + 1$ subsets

$$\left(l_0, l_0 + \frac{l_1 - l_0}{M+1}\right), \left(l_0 + \frac{l_1 - l_0}{M+1}, l_0 + \frac{2(l_1 - l_0)}{M+1}\right), \ldots, \left(l_0 + \frac{M(l_1 - l_0)}{M+1}, l_1\right).$$

We shall denote the boundary points $l_0 + \frac{m(l-1-l_0)}{M+1} = y_m$ for $m \in \{1, 2, \ldots, M\}$. We can also find $x_m = g^{-1}(y_m)$, where

$$g^{-1}(y_m) = \inf\{x_m \mid g(x_m) = y_m, \forall x_m \in [k_0, k_1]\}.$$

We then let number of transformations of Equation (2.11) be $T = M + 2$. The goal now to show that there exist a mapping $(a_t, b_t)_{t=1}^T = \mathcal{H}(\mathcal{S}_{ext})$, such that $|g_T(z_0) - g(z_0)| < \epsilon$ for all $z_0 \in [k_0, k_1]$.

Let $b_1 = 0$ and

$$a_1 = \begin{cases} \dfrac{\hat{\sigma}^{-1}(l_0)}{\sigma(k_0)} & \text{if } \sigma(k_0) \geq \hat{\sigma}^{-1}(l_0) \\ 1 & \text{otherwise.} \end{cases}$$

This ensures that for all $z_0 \in [k_0, k_1]$ we have $g_T(z_0) \geq l_0$. We then uses the next $M$ transformations to create a flow that maps $g_T(x_m) = y_m$. This is done by handling one and one $x_m$, while setting the $b_t$ in a manner that ensures

we do not change the previous $x_{1:m-1}$, as well as not change $g_T(k_0) \geq l_0$. Quick reminder that applying the flow for $t \in \mathcal{T}$ will first apply the standard logistic function to input, as $f_0(z_0) = \sigma(z_0)$. We can then define the parameters $(a_t, b_t)_2^{M+1}$ iteratively, for $t \in \{2, \ldots, M+1\}$ and letting $m = t-1$, as

$$b_t = \begin{cases} \hat{\sigma}^{-1}(y_{m-1}) & \text{if } t > 2 \\ \hat{\sigma}^{-1}(l_0) & \text{otherwise} \end{cases}$$

and

$$a_t = \frac{\hat{\sigma}^{-1}(y_m) - b_t}{\left(\bigcirc_{j=0}^{t-1} f_j(x_m)\right) - b_t}.$$

The transformations from $j = 0$ to $t-1$ are parameterized with the parameters already found, hence the iterative definition. Also, notice that the parameters does not use the input of $g_T$, they are constant. That is, after we have set a target $g$ and number of transformations $T$, we have a mapping which is constant w.r.t input. To set it in a NF perspective, the choice of $T$ is always set before approximating $\mathcal{H}$ and target $g$ is induced by the target distribution, which is also "set" before approximating $\mathcal{H}$, hence it is not dependent on the input to the flow, $z_0$.

After $M+1$ transformations using Equation (2.11) with the parameters described above, we simply need on last transformation. By setting

$$b_T = \hat{\sigma}^{-1}(y_M),$$

and

$$a_T = \begin{cases} \dfrac{\hat{\sigma}^{-1}(l_1) - b_T}{\left(\bigcirc_{j=0}^{T-1} f_j(k_1)\right) - b_T}, & \text{if } \hat{\sigma}^{-1}(l_1) \leq \bigcirc_{j=0}^{T} f_j(k_1) \\ 1, & \text{otherwise.} \end{cases}$$

Here we use the same logic as with the first transformations, but now ensuring that $g_T(z_0) \leq l_1$. We now have the final transformation with the property that for all $x_m$ with $m \in \{1, \ldots, M\}$, we have

$$g_T(x_t) = y_t.$$

To show convergence of $g_T$ and $g$, we simply see that $g_T(x_m) - g_T(x_{m-1}) = \frac{l_1 - l_0}{M+1}$ for all $m \in \{1, \ldots, M\}$ and $g_T(x_1) - l_0 = l_1 - g_T(x_M) = \frac{l_1 - l_0}{M+1}$. We also know that $l_0 \leq g_T(z_0) \leq l_1$, for all $z_0 \in [k_0, k_1]$. Using the fact that both functions $g_T$ and $g$ are monotonically increasing and $l_1 - l_0 \leq 1$, means that for all $z_0 \in [k_0, k_1]$,

$$|g_T(z_0) - g(z_0)| \leq \frac{l_1 - l_0}{M+1} < \frac{l_1 - l_0}{M} \leq \frac{1}{M} = \frac{1}{\lceil \frac{1}{\epsilon} \rceil} \leq \epsilon$$

As $D = 1$ means that the mapping must be constant for all $x \in [k_0, k_1]$, as $\mathcal{S}_{ext}(t, 1) = \emptyset$. This also means that the function $\mathcal{H}$ is continuous with input from a compact set, hence we can approximate it arbitraily well with a neural network following Theorem 1.0.2. ■

We now wish to extend the results above for higher dimensions, while also including the IAR-structure. We define a new function for each $d \in \mathcal{D}$, $G_d(z_{0,d}, \boldsymbol{z}_{0,1:d-1})$, where $\boldsymbol{z}_0 \in [k_0, k_1]^D$. When $\boldsymbol{z}_{1:d-1}$ is fixed, the function $G_d$ is a *strictly* monotonically increasing function w.r.t $z_{0,d}$, where $l_0^d \leq G_d(z_{0,d}, \boldsymbol{z}_{0,1:d-1}) \leq l_1^d$, with $0 \leq l_0^d < l_1^d \leq 1$. It is also continuous w.r.t $\boldsymbol{z}_{0,1:d-1}$, i.e given $\boldsymbol{z}_{0,1:d-1}$, for all $\epsilon > 0$ there exist a $\delta > 0$ such that

{eq: continG}

$$\begin{aligned} &||\boldsymbol{z}_{0,1:d-1} - \tilde{\boldsymbol{z}}_{0,1:d-1}||_\infty < \delta \\ &\implies |G_d(z_{0,d}, \boldsymbol{z}_{0,1:d-1}) - G_d(z_{0,d}, \tilde{\boldsymbol{z}}_{0,1:d-1})| < \epsilon, \end{aligned} \tag{2.16}$$

where $\tilde{z}_{0,1:d-1} \in [k_0, k_1]^{d-1}$. In addition to this, we allow for the end points $l_0^d, l_1^d$ to change when $z_{0,1:d-1}$ changes. However, they must be between 0 and 1 incuding, and $l_0^d < l_1^d$ and due to the continuity, the changes must be continuous as well. To be more precise, there must be continuity for the point $k_0$, i.e $G_d(k_0, z_{0,1:d-1}) = l_0^d$, and equivalently for $k_1, l_1^d$. To quickly summarize the properties of $G_d$:

- For each $\boldsymbol{z}_{0,1:d-1}$:
  - $\exists\, l_0^d, l_1^d \in [0,1]$ such that $l_0^d < l_1^d$ and $G_d \colon [k_0, k_1] \to [l_0^d, l_1^d]$.
  - $G_d$ is *strictly* monotonically increasing.
  - $G_d(k_0, \boldsymbol{z}_{0,1:d-1}) = l_0^d$ and $G_d(k_1, \boldsymbol{z}_{0,1:d-1}) = l_1^d$
- $G_d$ is continuous w.r.t $\boldsymbol{z}_{0,1:d-1}$, fulfilling Equation (2.16).
- The boundary points in the image of $G_d$ may change when $\boldsymbol{z}_{1:d-1}$ changes, but according to the points above, must do so continuously, with regards to Equation (2.16).

We wish to show we can converge towards $G = (G_1, G_2, \ldots, G_D)$ using PAF, similarly as in Lemma 2.6.10, by using the aformentioned lemma. However, we first need to make sure $(a_t, b_t)_{t=1}^T = \mathcal{H}_d(\boldsymbol{z}_{1:d-1})$, specified in the proof of Lemma 2.6.10, are continuous w.r.t $\boldsymbol{z}_{1:d-1}$. This to confirm we can approximate $\mathcal{H}_d$ arbitraily well using a neural network.

lemma: contin_ param_ paf

**Lemma 2.6.11.** *Let $(\mathcal{A}, g_T)$ be a PAF without the $T+2$ tranformation, with an IAR-structure. Let the function $G$ be defined as the function where each of the $D$ outputs is defined by $G_d$, i.e $G = (G_1, G_2, \ldots, G_D)$. Then there exist a flow on the form $g_T$, with continuity in $\mathcal{H}_d(\boldsymbol{z}_{0,1:d-1})$ for all $d \in \mathcal{D}$, which converges uniformly to $G$.*

*Proof.* For all $\boldsymbol{z}_{0,1:d-1}$ and for all $\epsilon > 0$, setting $M = \frac{1}{\lceil \epsilon \rceil}$ and letting $\mathcal{H}_d$ output the parameters specified in the proof of Lemma 2.6.10, gives uniform convergency to $G_d$ due to Lemma 2.6.11. Using the same $M$ for all $d \in \mathcal{D}$ and designing $\mathcal{H}_d$ as mentioned, gives uniform convergence for all $G_d$'s, hence there exist a flow $(\mathcal{A}, g_T)$ which converges uniformly towards $G$. We therefore only need to show continuity in $\mathcal{H}_d$ as described, for all $d \in \mathcal{D}$.

We show continuity for an arbitrary $d \in \mathcal{D} \setminus \{1\}$, see Lemma 2.6.10 for continuity when $d = 1$. Let $\epsilon_1 > 0$. For all $\epsilon_2 > 0$ there exist a $\delta_2 > 0$ such that that whenever

$$||\boldsymbol{z}_{0,1:d-1} - \tilde{\boldsymbol{z}}_{0,1:d-1}||_\infty < \delta_2$$

implies $|l_0^d - \tilde{l}_0^d| < \epsilon_2$, and equvalent argument for $l_1^d$, due to continuity in $G_d$. This means we can choose $\delta_2$ such that $|y_m - \tilde{y}_m| < 2\,\epsilon_2$. Combine this with the fact that $\hat{\sigma}^{-1}$ is continuous, means we can choose $\epsilon_2$ such that $|\hat{\sigma}^{-1}(y_m) - \hat{\sigma}^{-1}(\tilde{y}_m)| < \epsilon_1$ for all $m \in \{1, 2, \ldots, M\}$. Setting then $\delta_1 = \delta_2$ gives us

$$||\boldsymbol{z}_{0,1:d-1} - \tilde{\boldsymbol{z}}_{0,1:d-1}||_\infty < \delta_1 \implies ||\boldsymbol{b} - \tilde{\boldsymbol{b}}||_\infty < \epsilon_1,$$

where $\boldsymbol{b}$ and $\tilde{\boldsymbol{b}}$ are vectors with the correpsonding $(b_t)_{t=1}^T$ for $G_d(z_{0,d}, \boldsymbol{z}_{0,1:d-1})$ and $G_d(z_{0,d}, \tilde{\boldsymbol{z}}_{0,1:d-1})$ respectively.

Moving onto the $a$'s. Above shows that $\hat{\sigma}^{-1}(l_0)$ is continuous w.r.t $\boldsymbol{z}_{0,1:d-1}$. As

$$\lim_{\hat{\sigma}^{-1}(l_0) \to \sigma(k_0)} \frac{\hat{\sigma}^{-1}(l_0)}{\sigma(k_0)} = 1$$

means that $a_1$ is continuous. Shown above, we have that $y_m$ is continuous, and since $G_d$ is continuous and strictly increasing w.r.t $\boldsymbol{z}_{0,d-1}$, means that $x_m = G_d^{-1}(y_m, z_{0,1:d-1})$ is continuous for all $m \in \{1, \ldots, M\}$. As we already know $\hat{\sigma}^{-1}(y_1)$, $b_2$ and $\sigma(x_1)$ are all continuous, means

$$a_2 = \frac{\hat{\sigma}^{-1}(y_1) - b_2}{\sigma(x_1) - b_2}$$

is continuous. It follows inductively that $(a_t)_{t=3}^{M+1}$ are continuous, as $\bigcirc_{j=0}^{t-1} f_j(x_m)$ is continuous due to the fact that $x_m$, the standard logistic function, and all paramaters $(a_j, b_j)_{j=1}^{t-1}$ are continuous. Hence

$$a_t = \frac{\hat{\sigma}^{-1}(y_m) - b_t}{(\bigcirc_{j=0}^{t-1} f_j(x_m)) - b_t}$$

is continuous. Finally, with the same argument line we have that

$$\frac{\hat{\sigma}^{-1}(l_1) - b_T}{(\bigcirc_{j=0}^{T-1} f_j(k_1)) - b_T}$$

is continuous, and

$$\lim_{\hat{\sigma}^{-1}(l_1) \to \bigcirc_{j=0}^{T-1} f_j(k_1)} \frac{\hat{\sigma}^{-1}(l_1) - b_T}{(\bigcirc_{j=0}^{T-1} f_j(k_1)) - b_T} = 1$$

means that $a_T$ is continuous. Therefore, for any $\boldsymbol{z}_{0,1:d-1}$ and for all $\epsilon > 0$, we can find a $\delta > 0$ for each parameter (then simply pick the smallest $\delta$ of them), such that

$$||\boldsymbol{z}_{0,1:d-1} - \tilde{\boldsymbol{z}}_{0,1:d-1}||_\infty < \delta \implies ||\mathcal{H}_d(z_{0,1:d-1}) - \mathcal{H}_d(\tilde{z}_{0,1:d-1})||_\infty < \epsilon$$

As $d$ was arbitrarily chosen, and $d = 1$ is covered by Lemma 2.6.10, means it holds for all $d \in \mathcal{D}$ and hence $\mathcal{H}$ is continuous. ■

lemma: b_contin

**Lemma 2.6.12.** *Let $(\mathcal{A}, f)$ be a PAF with continuous parameter function $\mathcal{H}_d$. Then, for all $d \in \mathcal{D}$, the flow $f_d$ is continuous w.r.t $b_t$ for all $t \in \mathcal{T}$.*

*Proof.* We only focus showing continuity for $h_t$ for one $t$, as the identity function of $b_t$ is continuous, as well as we have preservation of continuity when it comes to addition and function composition. What we are going to show holds is the following. For every $z_d$, for each $\epsilon > 0$, and for each $a$, there exist a $\delta > 0$, namely $\delta = \epsilon/2a$, such that

$$|b - \tilde{b}| < \delta \implies |h_d(z_d - b_d) - h_d(z_d - \tilde{b}_d)| < \epsilon.$$

To show that this is true, we consider four different cases.

Case 1: Consider when both $z_d - b_t > 0$ as well as $z_d - (b_t \pm \delta) > 0$ (obviously it might only hold for $+\delta$, in which case we only consider that one). Then we have

$$|a_t(z_d - b_t) - a_t(z_d - (b_t \pm \delta))| = |\pm\delta| = \frac{\epsilon}{2a} < \epsilon,$$

where we use $\delta = \epsilon/2a$.

Case 2: Consider when both $z_d - b_t \leq 0$ and $z_d - (b_t \pm \delta) \leq 0$. Then we have

$$|(z_d - b_t) - (z_d - (b_t \pm \delta))| = |\pm\delta| = \frac{\epsilon}{2a} < \epsilon.$$

Case 3: Consider when $z_d - b_t > 0$ and $z_d - (b_t + \delta) \leq 0$, which also means $b_t < z_d \leq (b_t + \delta)$. We then have

$$|a_t(z_d - b_t) - (z_d - (b_t + \delta))| = |z_d(a_t - 1) - b_t(a_t - 1) + \delta|.$$

We can here consider three subcases. The first is when $a_t = 1$, then obviously have

$$|z_d(a_t - 1) - b_t(a_t - 1) + \delta| < |\delta| = \frac{\epsilon}{2a} < \epsilon.$$

If $a_t > 1$, we have, keeping in mind the bounds on $z_d$, we have

$$|z_d(a_t - 1) - b_t(a_t - 1) + \delta| \leq |(b_t + \delta)(a_t - 1) - b_t(a_t - 1) + \delta| = |a_t\delta| = \frac{\epsilon}{2} < \epsilon.$$

And finally, if $a_t < 1$, we have

$$|z_d(a_t - 1) - b_t(a_t - 1) + \delta| < |b_t(a_t - 1) - b_t(a_t - 1) + \delta| = |\delta| = \frac{\epsilon}{2a} < \epsilon.$$

Case 4: Finally, consider when $z_d - b_t \leq 0$, while $z_d - (b_t - \delta) > 0$, which gives us the bounds $(b_t - \delta) < z_d \leq b_t$. We then have

$$|(z_d - b_t) - a_t(z_d - (b_t - \delta))| = |z_d(1 - a_t) - b_t(1 - a_t) - a_t\delta|.$$

Considering again three subcases. When $a_t = 1$, we have

$$|z_d(1 - a_t) - b_t(1 - a_t) - a_t\delta| = |-a_t\delta| = \frac{\epsilon}{2} < \epsilon.$$

When $a_t > 1$ we have, keeping in mind the boundaries given above,

$$|z_d(1 - a_t) - b_t(1 - a_t) - a_t\delta| \leq |b_t(1 - a_t) - b_t(1 - a_t) - a_t\delta| = |-a_t\delta| = \frac{\epsilon}{2} < \epsilon.$$

And finally, when $a_t < 1$, we have

$$|z_d(1-a_t) - b_t(1-a_t) - a_t\delta| < |(b_t-\delta)(1-a_t) - b_t(1-a_t) - a_t\delta| = |-\delta| = \frac{\epsilon}{2a} < \epsilon.$$

Hence, $h_t(z_d - b_t)$ is continuous w.r.t $b_t$ for all $t \in \mathcal{T}$ and $d \in \mathcal{D}$. By the argument in the start of the proof, it follows that $f_d$ is continuous w.r.t $b_t$ for all $t \in \mathcal{T}$ and $d \in \mathcal{D}$. ■

We can now show convergence to $G$ of the flow $(\mathcal{A}, g_T)$ which approximates $\mathcal{H}$ with neural networks.

lemma: paf_ multi_ dim_NN

**Lemma 2.6.13.** *Let $(\mathcal{A}, g_T)$ be a PAF without the T+2 transformation, with an IAR structure and for each $d \in \mathcal{D}$, $\mathcal{H}_d$ is approximated by a neural network $\mathcal{NN}^{\phi}_{d+3}$ with $k_d$ number of hidden layers. Then there exist a flow of $(\mathcal{A}, g_T)$ that converges toward $G$ as $T \to \infty$ and $K_d \to \infty$.*

*Proof.* We start by showing convergence for an arbitrary $d \in \mathcal{D}$. Firstly, as the input to $\mathcal{H}_d$ is compact and the function itself is continuous as seen in Lemma 2.6.11, means that we can for any $\delta > 0$ find a $K_d$ such that whenever $k_d > K_d$ implies

$$||\mathcal{NN}^{\phi}_{d+3}(\boldsymbol{z}_{0,1:d-1}) - \mathcal{H}_d(\boldsymbol{z}_{0,1:d-1})||_\infty < \delta,$$

due to Theorem 1.0.2.

Let $g_{T,d}(z_{0,d}; \mathcal{H}_d)$ be the dth transformation of the NF, using $\mathcal{H}_d$ to output the parameters, and equivalently $g_{T,d}(z_{0,d}; \mathcal{NN}^{\phi}_{d+3})$. We now wish to show convergence between these two. For this we first show that $g_{T,d}$ is uniformly continuous w.r.t to its parameters $(a_t, b_t)_{t=1}^T$), regardless of using $\mathcal{NN}^{\phi}_{d+3}$ or $\mathcal{H}_d$. The line of arguments goes as follows:

1. The derivatives of $g_{T,d}$ w.r.t $a_t$ for $t \in \mathcal{T}$,

$$\frac{\partial g_{T,d}}{\partial a_t} = \begin{cases} z_{t,d} - b_t, & \text{if } z_{t,d} - b_t > 0 \\ 0, & \text{otherwise.} \end{cases}$$

2. As the derivatives always exist for all $a_t$ and from Lemma 2.6.12, means $g_{T,d}$ is continuous w.r.t its parameters, where $d \in \mathcal{D}$.

3. The input to $\mathcal{H}_d/\mathcal{NN}^{\phi}_{d+3}$ is, by definition, compact. Both functions are continuous, $\mathcal{H}_d$ due to Lemma 2.6.11 and $\mathcal{NN}^{\phi}_{d+3}$ follows from how we defined it in Definition 1.0.1.

4. Compactness and continuity of function implies compactness w.r.t output, hence the input to $g_{T,d}$ w.r.t the paramaters is compact.

5. Continuity in $g_{T,d}$ w.r.t the parameters combined with the fact that the input is compact, implies uniform convergence w.r.t $(a_t, b_t)_{t=1}^T$, as $d$ was arbitrary, means it also holds for all $d \in \mathcal{D}$.

Uniform convergence combined with convergence of the network, means there exist for all $z_{0,d} \in [k_0^d, k_1^d]$ and for every $\epsilon/2 > 0$ a $\delta > 0$ (by picking $K_d$ large enough), such that

$$\begin{aligned} &||\mathcal{NN}^{\phi}_{d+3}(\boldsymbol{z}_{0,1:d-1}) - \mathcal{H}_d(\boldsymbol{z}_{0,1:d-1})||_\infty < \delta \\ &\implies |g_{T,d}(z_{0,d}; \mathcal{NN}^{\phi}_{d+3}) - g_{T,d}(z_{0,d}; \mathcal{H}_d)| < \frac{\epsilon}{2} \end{aligned}$$

By combining this with Lemma 2.6.11, shows that for all $z_{0,1:d} \in [k_0, k_1]^d$ and for every $\epsilon > 0$, there exist a $T_d$ and $K_d$ such whenever $T > T_d$ and $k_d > K_d$, we have

$$\begin{aligned} &|g_{T,d}(z_{0,d}; \mathcal{NN}^{\phi}_{d+3}) - G_d(z_{0,d}, z_{0,1:d-1})| \\ &\qquad \leq |g_{T,d}(z_{0,d}; \mathcal{NN}^{\phi}_{d+3}) - g_{T,d}(z_{0,d}; \mathcal{H}_d)| + \\ &\qquad\quad |g_{T,d}(z_{0,d}; \mathcal{H}_d) - G_d(z_{0,d}, z_{0,1:d-1})| \\ &\qquad < \frac{\epsilon}{2} + \frac{\epsilon}{2} = \epsilon \end{aligned}$$

For all $\boldsymbol{z}_0 \in [k_0, k_1]^D$ and for every $\epsilon > 0$ there must exist an

$$U = \{U \mid U \in \mathcal{D} \text{ and } T_U = \sup\{T_d\}_{d=1}^D\}$$

and a

$$V = \{V \mid V \in \mathcal{D} \text{ and } K_V = \sup\{K_d\}_{d=1}^D\}$$

such that $(k_d > V)_{d=1}^D$ and $T > U$, we have

$$||g_T(\boldsymbol{z}_0; \mathcal{NN}^{\phi}_{d+3}) - G(\boldsymbol{z}_0)||_\infty < \epsilon,$$

where $g_T(\boldsymbol{z}_0; \mathcal{NN}^{\phi}_{d+3})$ is the NF with a neural network for each dimension, following the Definition 1.0.1 and each with $k_d$ depth. ■

We now define the class of distributions we are proving that PAF with IAF-structure is an UDA for.

def: strict_ condit_ fam

**Definition 2.6.14.** Let $\mathcal{X} \subseteq \mathbb{R}^D$ be a connected subset. The *strictly conditionally continuous*-family is the class of densities $\mathscr{P}$ such that the density $p_{\boldsymbol{x}} \in \mathscr{P}$ follows

$$\begin{cases} p_{\boldsymbol{x}}(\boldsymbol{x}) > 0, & \text{if } \boldsymbol{x} \in \mathcal{X} \\ p_{\boldsymbol{x}}(\boldsymbol{x}) = 0, & \text{if } \boldsymbol{x} \notin \mathcal{X}. \end{cases}$$

Also, the conditional CDF of the density, $F_d(x_d \mid \boldsymbol{x}_{1:d-1})$, is continuous w.r.t $\boldsymbol{x}_{1:d-1}$.

The class used in Theorem 2.6.2, regarding UDA of NAF-DSF, is obviously contained in the class above. However this one is larger as we allow for both discontinuities in the density and also the density to be 0 at some subset of $\mathbb{R}^D$. Before we can prove the main result in this section, we need to include an important lemma shown by Huang et al. 2018.

lemma: huang_ conv

**Lemma 2.6.15** (Lemma 4, Huang et al. 2018)**.** *Let $\mathcal{Z} \subseteq \mathbb{R}^D$ and $\mathcal{X} \subseteq \mathbb{R}^D$, with each bein the sample space of a probability space, i.e $(\mathcal{Z}, \mathscr{B}(\mathcal{Z}), \mu)$ and $(\mathcal{X}, \mathscr{B}(\mathcal{X}), \nu)$. Let $J \colon \mathcal{Z} \to \mathcal{X}$ be any function and $J_n$ be a sequence of functions such that $J_n$ converges pointwise to $J$. Then a transformation of the form $x_n = J_n(z)$ converges in distribution to $x = J(z)$.*

The proof is quite straightforward by introducing a bounded continuous function $h$, and show convergence in expectation of $h(z_n)$ to $h(z)$ by dominated convergence theorem. Then simply finish it by applying the Portmanteau's lemma.

thm: paf_uda_ multi

**Theorem 2.6.16.** *Let $(\mathcal{A}, f)$ be a PAF with IAR-structure with, for each $d \in \mathcal{D}$, the paramaters are computed by a neural network $\mathcal{NN}^{\phi}_{d+3}$ with arbitrarily depth. Let $\mathscr{P}$ be the class of densities defined as strictly conditionally continuous. There exist flow $f$ for every distribution $p_{\boldsymbol{x}}$ in $\mathscr{P}$ such that when the number of transformations $T \to \infty$, $f(\boldsymbol{z}_0) \xrightarrow{d} \boldsymbol{x}$*

*Proof.* Let $F \colon \mathbb{R}^D \to [0,1]^D$ with the $d$th output defined as the conditional CDF to $p_{\boldsymbol{x}}$, i.e $\hat{x}_d = F_d(x_d \mid x_{1:d-1}) = Pr(X_d < x_d \mid x_{1:d-1})$. With $\boldsymbol{x} \in [0,1]^D$. When we are working with $\boldsymbol{x}_{1:d-1}$, we are implictly restricting possible values such that $p_{\boldsymbol{x}}(x_d \mid \boldsymbol{x}_{1:d-1}) > 0$ for some $x_d \in \mathbb{R}$. Due to $\mathscr{P}$, we have that the set of possible values $\boldsymbol{x}_{1:d-1}$ is a connected subset of $\mathbb{R}^{d-1}$, hence when we have a $\boldsymbol{x}_{1:d-1}$ and talk about $||\boldsymbol{x}_{1:d-1} - \tilde{\boldsymbol{x}}_{1:d-1}||_\infty$ we talk about the set which fulfill the inequality and also are possible values. They in themselves comprise of a connected subspace which is never empty nor only $\boldsymbol{x}_{1:d-1}$. Going forward we are implictly adding this restriction.

When $x_{1:d-1}$ is fixed, we have two numbers $l_0^d < l_1^d$ (we allow for $\pm\infty$), such that it is strictly monotonically increasing when $x_d \in [l_0^d, l_1^d]$ (obviously the set is open when $\pm\infty$) per the requirement of stricly positive density, 0 when $x_d < l_0^d$ and 1 when $x_d > l_1^d$. We can also see, due to continuity in the conditional, that the boundary points when restricted to the strictly monotonically increasing part can change, but only continuously in the same manner as with $G_d$. Think of it as the part that is 0 and 1 in the $F_d$ can only change slightly and only the part that is close to the stricly increasing part, otherwise we break the continuity of $F_d$ w.r.t $\boldsymbol{x}_{1:d-1}$.

Let $F_d^{-1}(\hat{x}_d \mid \boldsymbol{x}_{1:d-1})$ be defined as the inverse of $F_d(x_d \mid \boldsymbol{x}_{1:d-1})$, where the image of the inverse is simply the values mapping to the strictly increasing part. We call this interval for $I \subseteq \mathbb{R}$, so $F_d \colon I \to [0,1]$ is strictly increasing. We now show continuity for the inverse w.r.t both $\hat{x}_d$, and also w.r.t $\boldsymbol{x}_{1:d-1}$. When $\boldsymbol{x}_{1:d-1}$ is fixed, means the inverse $F_d^{-1}$ is continuous w.r.t $x_d$, as $F_d$ restricted to the interval that is the image of $F_d^{-1}$ is strictly increasing and continuous. This is easy to see, as for any $\epsilon > 0$ and any $x_d \in I$, we have

$$F_d(x_d - \epsilon \mid \boldsymbol{x}_{1:d-1}) < F_d(x_d \mid \boldsymbol{x}_{1:d-1}) < F_d(x_d + \epsilon \mid \boldsymbol{x}_{1:d-1}),$$

which by setting $\delta$ to be the minimum of $|F_d(x_d \pm \epsilon \mid \boldsymbol{x}_{1:d-1} - F_d(x_d \mid \boldsymbol{x}_{1:d-1})|$ (some small minor details when $x_d$ is a boundary point in $I$ or if $x_d \pm \epsilon \notin I$, however this is easy to handle by considering left/right continuity in the boundary case and simply picking some points closer towards $F_d(x_d \mid \boldsymbol{x}_{1:d-1})$ in the second case).

Next we look at continuity w.r.t $\boldsymbol{x}_{1:d-1}$. Let $\epsilon > 0$ and for any $\boldsymbol{x}_{1:d-1}$ we have the following. Let $\delta_1 > 0$ be set so that

$$|\hat{x}_d - \hat{x}'_d| < \delta_1 \implies |F_d^{-1}(\hat{x}_d \mid \boldsymbol{x}_{1:d-1}) - F_d^{-1}(\hat{x}'_d \mid \boldsymbol{x}_{1:d-1})| < \epsilon.$$

Using continuity in conditional CDF, we can find $\delta_2 > 0$ such that whenever $||\boldsymbol{x}_{1:d-1} - \tilde{\boldsymbol{x}}_{1:d-1}||_\infty < \delta_2$ we have

$$|F_d(x_d \mid \boldsymbol{x}_{1:d-1}) - F_d(x_d \mid \tilde{\boldsymbol{x}}_{1:d-1})| < \delta_1. \tag{2.17}$$

{eq: delta1}

Let

$$\tilde{X} = \{\tilde{\boldsymbol{x}}_{1:d-1} \mid ||\tilde{\boldsymbol{x}}_{1:d-1} - x_{1:d-1}||_\infty < \delta_2\}$$

a mapping $\eta \colon \tilde{X} \to (0, 1]$ defined as

$$\eta(\tilde{\boldsymbol{x}}_{1:d-1}) = \max\{\delta \mid 0 < \delta \leq 1 \text{ and } \forall \hat{x}'_d : |\hat{x}_d - \hat{x}'_d| < \delta \implies |F_d^{-1}(\hat{x}_d \mid \tilde{\boldsymbol{x}}_{1:d-1}) - F_d^{-1}(\hat{x}'_d \mid \tilde{\boldsymbol{x}}_{1:d-1})| < \epsilon\}.$$

This mapping simply take the largest $\delta$ that fulfills continuity w.r.t $\hat{x}_d$ or 1, if the value can be larger than one. We know at least one such $\delta$ exist, as we know there continuity when what we condition on is fixed. Let then $\delta_3 > 0$ be defined as

$$\delta_3 = \min\{\delta \mid \tilde{\boldsymbol{x}}_{1:d-1} \in \tilde{X} \text{ and } \delta = \eta(\tilde{\boldsymbol{x}}_{1:d-1})\}$$

and set $\delta_4$ equivalently to how we set $\delta_2$ using Equation (2.17), but replacing $\delta_1$ with $\delta_3$. For any $\tilde{\boldsymbol{x}}_{1:d-1}$, let $\hat{x}_d = F_d(x_d \mid \boldsymbol{x}_{1:d-1})$ and $\hat{x}'_d = F_d(x_d \mid \tilde{\boldsymbol{x}}_{1:d-1})$, then whenever $|\boldsymbol{x}_{1:d-1} - \tilde{\boldsymbol{x}}_{1:d-1}| < \delta_4$ we have

$$\begin{aligned}
&|F_d^{-1}(\hat{x}_d \mid \boldsymbol{x}_{1:d-1}) - F_d^{-1}(\hat{x}_d \mid \tilde{\boldsymbol{x}}_{1:d-1})| \\
\leq &|F_d^{-1}(\hat{x}_d \mid \boldsymbol{x}_{1:d-1}) - F_d^{-1}(\hat{x}'_d \mid \tilde{\boldsymbol{x}}_{1:d-1})| + |F_d^{-1}(\hat{x}'_d \mid \tilde{\boldsymbol{x}}_{1:d-1}) - F_d^{-1}(\hat{x}_d \mid \tilde{\boldsymbol{x}}_{1:d-1})| \\
= &0 + |F_d^{-1}(\hat{x}'_d \mid \tilde{\boldsymbol{x}}_{1:d-1}) - F_d^{-1}(\hat{x}_d \mid \tilde{\boldsymbol{x}}_{1:d-1})|.
\end{aligned}$$

Due to continuity w.r.t $\boldsymbol{x}_{1:d-1}$, we have $|\hat{x}_d - \hat{x}'_d| < \delta_3$, hence we have

$$|F_d^{-1}(\hat{x}'_d \mid \tilde{\boldsymbol{x}}_{1:d-1}) - F_d^{-1}(\hat{x}_d \mid \tilde{\boldsymbol{x}}_{1:d-1})| < \epsilon.$$

If we now apply the standard logistic function, $\sigma(F_d^{-1}(\hat{x}_d \mid \boldsymbol{x}_{1:d-1}))$, we have something very close to $G_d$, but the domain is different. If we now construct a function $G_d$ as a composition of the following two steps, with $\boldsymbol{z}_{0,1:d} \in [k_0, k_1]^d$,

$$\begin{aligned}
\hat{\boldsymbol{z}}_{0,1:d} &= \frac{\sigma(\boldsymbol{z}_{0,1:d}) - \sigma(k_0)}{\sigma(k_1)} \\
&\sigma \circ F_d^{-1}(\hat{z}_{0,d} \mid \hat{\boldsymbol{z}}_{0,1:d-1}),
\end{aligned}$$

where everything is elementwise on the first line. Letting $G = (G_1, G_2, \ldots, G_D)$, we know from Lemma 2.6.13 that there exist a flow $g_T$ with IAR-structure and approximating $\mathcal{H}$ with neural networks, such that it converges uniformly to $G$. As uniform convergence implies pointwise and adding the logit function to $G$ and $g_T$, which preserves continuity, means $\boldsymbol{z}_T = f(\boldsymbol{z}_0) = \sigma^{-1}(g_T(\boldsymbol{z}_0))$ converges to $\sigma^{-1}(G(\boldsymbol{z}_0))$. Hence, due to Lemma 2.6.15, $\boldsymbol{z}_T \xrightarrow{d} \boldsymbol{x}$. ■

Lets now consider the case when $D = 1$, in which any flow with affine transformations failed be an UDA for any meaningful class of distributions. It turns out that in this case one can show that PAF is and UDA for an even larger class of distribution. The class of distributions is the set of all continuous distributions. To show this we quickly introduce another flow, quite similar to PAF.

$$\hat{\sigma} \circ \bigcirc_{t=1}^{T_1} f_t \circ \hat{\sigma}^{-1} \circ \sigma(x). \tag{2.18}$$

{eq: one_dim_ backward}

And a quick reminder of what PAF without $T + 2$ transformation looks like,

$$\hat{\sigma} \circ \bigcirc_{t=1}^{T_2} f_t \circ \sigma(z_0). \tag{2.19}$$

{eq: one_dim_ forward}

**Proposition 2.6.17.** *Let $(\mathcal{A}, f)$ be a PAF with $\mathcal{H}$ being approximated by a $\mathcal{NN}^{\phi}_{2T+3}$ and $D = 1$. Let $\mathscr{P}$ be the class of all possible densities. For any density $p_x \in \mathscr{P}$, there exist a flow $f$ such that when the number of transformations $T \to \infty$, $f(z_0) \xrightarrow{d} x$.*

*Proof.* With minor changes in the proof of Lemma 2.6.10, we can easily show the same result holds for Equation (2.18). Let $F_1(z_0)$ be the CDF of $\mathcal{A}$, and $F_2(x)$ is the CDF to $p_x \in \mathscr{P}$. From Lemma 2.6.10, we have that Equation (2.19) can arbitraily well approximate $F_1$, and equally with $F_2$ and Equation (2.18). Setting $T = T_1 + T_2$ and observing that the composition of Equation (2.19) and the inverse of Equation (2.18) gives a PAF. From Lemma 2.6.10, we know that we can find this approximation by letting the parameter-function $\mathcal{H}$ be a neural network. The composition approximate arbitrarily well $F_2^{-1} \circ F_1(z_0)$. Following Lemma 2.6.15 we have that $z_T \xrightarrow{d} x$. ■

This shows the effect of the non-linearity we discussed above. An effect that makes the smoking gun when it comes to show that affine transformations can be an UDA, is quite the opposite for PAF.

# Bibliography


naf [Hua+18] Huang, C.-W. et al. 'Neural Autoregressive Flows'. In: ed. by Dy, J. and Krause, A. Vol. 80. Proceedings of Machine Learning Research. Stockholmsmässan, Stockholm Sweden: PMLR, Oct. 2018, pp. 2078–2087.

iaf [Kin+16] Kingma, D. P. et al. 'Improved Variational Inference with Inverse Autoregressive Flow'. In: *Advances in Neural Information Processing Systems 29*. Ed. by Lee, D. D. et al. Curran Associates, Inc., 2016, pp. 4743–4751.

nn_uni [KL20] Kidger, P. and Lyons, T. 'Universal Approximation with Deep Narrow Networks'. In: ed. by Abernethy, J. and Agarwal, S. Vol. 125. Proceedings of Machine Learning Research. PMLR, Sept. 2020, pp. 2306–2327.

nf_review [KPB20] Kobyzev, I., Prince, S. and Brubaker, M. 'Normalizing Flows: An Introduction and Review of Current Methods'. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2020), pp. 1–1.

real_analysis [MW13] McDonald, J. N. and Weiss, N. *A course in real analysis*. 2nd ed. Elsevier Inc., 2013.

maf [PPM17] Papamakarios, G., Pavlakou, T. and Murray, I. 'Masked Autoregressive Flow for Density Estimation'. In: *Advances in Neural Information Processing Systems 30*. Ed. by Guyon, I. et al. Curran Associates, Inc., 2017, pp. 2338–2347.

scheffe [Sch47] Scheffe, H. 'A Useful Convergence Theorem for Probability Distributions'. In: *Ann. Math. Statist.* Vol. 18, no. 3 (Sept. 1947), pp. 434–438.

wehenkel [WL20] Wehenkel, A. and Louppe, G. 'You say Normalizing Flows I see Bayesian Networks'. In: *arXiv preprint arXiv:2006.00866v2* (2020).