

Learning Bijective Feature Maps for Linear ICA

Alexander Camuto^{*,1,2}

Matthew Willetts^{*,1,2}

Brooks Paige^{2,3}

Chris Holmes^{1,2}

Stephen Roberts^{2,4}

ACAMUTO@TURING.AC.UK

MWILLETTS@TURING.AC.UK

B.PAIGE@UCL.AC.UK

CHOLMES@STATS.OX.AC.UK

SJROB@ROBOTS.OX.AC.UK

¹*Department of Statistics, University of Oxford*

²*Alan Turing Institute, London*

³*Department of Computer Science, University College London*

⁴*Oxford-Man Institute, University of Oxford*

Abstract

Separating high-dimensional data like images into independent latent factors, i.e independent component analysis (ICA), remains an open research problem. As we show, existing probabilistic deep generative models (DGMs), which are tailor-made for image data, underperform on non-linear ICA tasks. To address this, we propose a DGM which combines bijective feature maps with a linear ICA model to learn interpretable latent structures for high-dimensional data. Given the complexities of jointly training such a hybrid model, we introduce novel theory that constrains linear ICA to lie close to the manifold of orthogonal rectangular matrices, the Stiefel manifold. By doing so we create models that converge quickly, are easy to train, and achieve better unsupervised latent factor discovery than flow-based models, linear ICA, and Variational Autoencoders on images.

1. Introduction

In linear Independent Component Analysis (ICA), data is modelled as having been created from a linear mixing of independent latent *sources* (Cardoso, 1989a,b, 1997; Comon, 1994; Bell & Sejnowski, 1995). The canonical problem is blind source separation; the aim is to estimate the original sources of a mixed set of signals by learning an *unmixing* matrix, which when multiplied with data recovers the values of these sources. While linear ICA is a powerful approach to undo the mixing of signals like sound (Everson & Roberts, 2001; Hyvärinen et al., 2001), it has not been as effectively developed for learning compact representations of high-dimensional data like images, where the linear assumption is limiting. Non-linear ICA methods, which assume the data has been created from a non-linear mixture of latent sources, offer better performance on such data.

In particular, flow-based models have been proposed as a non-linear approach to *square* ICA, where we assume the dimensionality of our latent source space is the same as that of our data (Deco & Brauer, 1995; Parra et al., 1995; Dinh et al., 2015, 2017). Flows parameterise a bijective mapping between data and a feature space of the same dimension and can be trained under a maximum likelihood objective for a chosen base distribution in that space. While these are powerful generative models, for image data one typically wants fewer latent variables than the number of pixels in an image. In such situations, we wish to learn a non-square (dimensionality-reducing) ICA representation.

In this work, we highlight the fact that existing probabilistic deep generative models (DGMs), in particular Variational Autoencoders (VAEs), underperform on non-linear ICA tasks. As such there is a real need for a probabilistic DGM that can perform these tasks. To address this we propose a novel methodology for performing non-square non-linear ICA using a model, termed *Bijecta*, with two jointly trained parts: a highly-constrained non-square linear ICA model, operating on a feature

*. Equal Contribution.

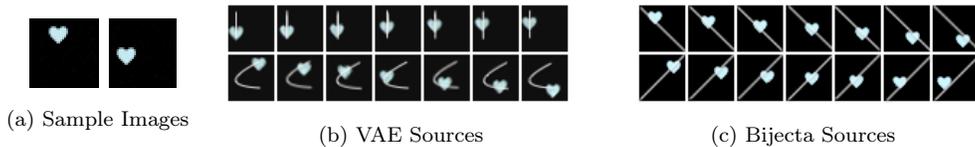


Figure 1: Here we take a dSprites heart and, using a randomly sampled affine transformation, move it around a black background (a). The underlying sources of the dataset are *affine* transformations of the heart. In (b-c) images in the center correspond to the origin of the learnt source space. Images on either side correspond to linearly increasing values along one of the learnt latent sources whilst the other source remains fixed. Bijecta (c) has learned affine transformations as sources (white diagonals), whereas a VAE (with ICA-appropriate prior) (b) has learned non linear transforms (white curves). The VAE has not discovered the underlying latent sources.

space output by a bijective flow. The bijective flow is tasked with learning a representation for which linear ICA is a good model. It is as if we are *learning the data* for our ICA model.

We find that such a model fails to converge when trained naively with no constraints. To ensure convergence, we introduce novel theory for the parameterisation of decorrelating, non-square ICA matrices that lie close to the Stiefel manifold (Stiefel, 1935), the space of orthonormal rectangular matrices. We use this result to introduce a novel non-square linear ICA model that uses Johnson-Lindenstrauss projections (a family of randomly generated matrices). Using these projections, Bijecta successfully induces dimensionality reduction in flow-based models and scales non-square non-linear ICA methods to high-dimensional image data. Further we show that it is better able to learn independent latent factors than each of its constituent components in isolation and VAEs. For a preliminary demonstration of the inability of VAEs and the ability of Bijecta to discover ICA sources see Fig 1.

2. Background

2.1 Independent Component Analysis

The goal of ICA is to learn a set of statistically independent sources that ‘explain’ our data. ICA is a highly diverse modelling paradigm with numerous variants: learning a mapping vs learning a model, linear vs non-linear, different loss functions, different generative models, and a wide array of methods of inference (Cardoso, 1989a; Mackay, 1996; Lee et al., 2000).

Here, we specify a generative model and find point-wise maximum likelihood estimates of model parameters in the manner of (Mackay, 1996; Cardoso, 1997). Concretely, we have a model with latent sources $\mathbf{s} \in \mathcal{S} = \mathbb{R}^{d_s}$ generating data $\mathbf{x} \in \mathcal{X} = \mathbb{R}^{d_x}$, with $d_s \leq d_x$. The linear ICA generative model factorises as

$$p(\mathbf{x}, \mathbf{s}) = p(\mathbf{x}|\mathbf{s})p(\mathbf{s}), \quad p(\mathbf{s}) = \prod_{i=1}^{d_s} p(s_i),$$

where $p(\mathbf{s})$ is a set of independent distributions appropriate for ICA. In linear ICA, where all mappings are simple matrix multiplications, the sources *cannot* be Gaussian distributions. Recall that we are mixing our sources to generate our data: A linear mixing of Gaussian random variables is itself Gaussian, so unmixing is impossible (Lawrence & Bishop, 2000). To be able to unmix, to break this symmetry, we can choose any heavy-tailed or light-tailed non-Gaussian distribution as our prior $p(\mathbf{s})$ that gives us axis alignment and independence between sources. A common choice is the family of

generalised Gaussian distributions,

$$p(s_i) = \text{GG}(s_i|\mu, \alpha, \rho) = \frac{\rho}{2\alpha\Gamma(1/\rho)} \exp\left[\left(-\frac{|s_i - \mu|}{\alpha}\right)^\rho\right] \quad (1)$$

with mean μ , scale α and shape ρ . For $\rho = 2$ we recover the Normal distribution, and for $\rho = 1$ we have the (heavy-tailed) Laplace. As $\rho \rightarrow \infty$ the distribution becomes increasingly sub-Gaussian, tending to a uniform distribution. As such, the generalised Gaussian is a flexible framework for specifying ICA-appropriate priors as it allows for the specification of a sub or super Gaussian distribution by way of a single parameter: ρ .

2.2 Manifolds for the unmixing matrix \mathbf{A}^+

In linear ICA we want to find the linear mapping \mathbf{A}^+ resulting in *maximally independent* sources. This is more onerous than merely finding decorrelated sources, as found by principal component analysis (PCA).

When learning a linear ICA model we typically have the mixing matrix \mathbf{A} as the (pseudo)inverse of the unmixing matrix \mathbf{A}^+ and focus on the properties of \mathbf{A}^+ to improve convergence. \mathbf{A}^+ linearly maps from the data-space \mathcal{X} to the source space \mathcal{S} . It can be decomposed into two linear operations. First we *whiten* the data such that each component has unit variance and these components are mutually uncorrelated. We then apply an orthogonal transformation and a scaling operation (Hyvärinen et al., 2001, §6.34) to ‘rotate’ the whitened data into a set of coordinates where the sources are independent *and* decorrelated. Whitening on its own is not sufficient for ICA — two sources can be uncorrelated *and* dependent (see Appendix A).

Thus we can write the linear ICA unmixing matrix as

$$\mathbf{A}^+ = \mathbf{\Phi}\mathbf{R}\mathbf{W} \quad (2)$$

where $\mathbf{W} \in \mathbb{R}^{d_s \times d_x}$ is our whitening matrix, $\mathbf{R} \in \mathbb{R}^{d_s \times d_s}$ is an orthogonal matrix and $\mathbf{\Phi} \in \mathbb{R}^{d_s}$ is a diagonal matrix. Matrices that factorise this way are known as the *decorrelating matrices* (Everson & Roberts, 1999): members of this family decorrelate through \mathbf{W} , and $\mathbf{\Phi}\mathbf{R}$ ensures that sources are statistically independent, not merely uncorrelated. The optimal ICA unmixing matrix is the decorrelating matrix that decorrelates *and* gives independence.

2.3 Flows

Flows are models that stack numerous invertible changes of variables. One specifies a simple base distribution and learns a sequence of (invertible) transforms to construct new distributions that assign high probability to observed data. Given a variable $\mathbf{z} \in \mathcal{Z} = \mathbb{R}^{d_x}$, we specify the distribution over data \mathbf{x} as

$$p(\mathbf{x}) = p(\mathbf{z}) \left| \det \frac{\partial f^{-1}}{\partial \mathbf{z}} \right|, \quad (3)$$

where f is a bijection from $\mathcal{Z} \rightarrow \mathcal{X}$, ie $\mathbb{R}^{d_x} \rightarrow \mathbb{R}^{d_x}$, and $p(\mathbf{z})$ is the base distribution over the latent \mathbf{z} (Rezende & Mohamed, 2015; Papamakarios et al., 2019).

For more flexible distributions for \mathbf{x} , we specify \mathbf{x} through a series of composed functions, from our simple initial p into a more complex multi-modal distribution; for example for a series of $K + 1$ mappings, $\mathbf{z} = f_K \circ \dots \circ f_0(\mathbf{x})$. By the properties of determinants under function composition

$$p(\mathbf{x}) = p(\mathbf{z}_K) \prod_{i=0}^K \left| \det \frac{\partial f_i^{-1}}{\partial \mathbf{z}_{i+1}} \right|, \quad (4)$$

where \mathbf{z}_i denotes the variable resulting from the transformation $f_i(\mathbf{z}_i)$, $p(\mathbf{z}_K)$ defines a density on the K^{th} , and the bottom most variable is our data ($\mathbf{z}_0 = \mathbf{x}$).

Computing the determinant of the Jacobian ($\det \frac{\partial f^{-1}}{\partial \mathbf{z}}$) in Eq. (3) can be prohibitively costly, especially when composing multiple functions as in Eq. (4). To address this, flows use *coupling layers* that enforce a lower triangular Jacobian such that the determinant of the Jacobian is simply the product of its diagonal elements. We use recently proposed coupling layers based on rational quadratic splines (RQS) to enforce this lower triangular structure (Durkan et al., 2019). They form highly flexible flows that typically require fewer composed mappings to achieve good performance relative to other coupling layers. See Appendix H for details.

3. Non-Square ICA using Flows

Variational Autoencoders seem like a natural fit for learning a compressed set of statistically independent latent variables (Kingma & Welling, 2014; Rezende et al., 2014). It seems natural to train a VAE with an appropriate non-Gaussian prior, and expect that it would learn an appropriate ICA model. However, this is not the case. In Khemakhem et al. (2020) some experiments suggest that VAEs with ICA-appropriate priors are unsuited to performing non-linear ICA. In our experiments (§5) we further verify this line of inquiry and show that VAEs struggle to match their aggregate posteriors to non-Gaussian priors and thus are unable to discover independent latent sources.

Though source separation can be achieved by ‘disentangling’ methods such as the β -VAE (Higgins et al., 2017) and β -TCVAE (Chen et al., 2018), these methods require post-hoc penalisation of certain terms of the VAE objective, at times inducing improper priors (in the β -TCVAE in particular (Mathieu et al., 2019)). Further, precise tuning of this penalisation, a form of soft supervision, is key to getting appropriate representations (Rolinek et al., 2019; Locatello et al., 2019). Stühmer et al. (2019) obtains a variety of non-linear ICA using VAEs with sets of Generalised Gaussian priors, but even then β penalisation is required to obtain ‘disentangled’ representations.

As such there is a need for probabilistic DGMs that can separate sources without added hyperparameter tuning and that can do so by matching ICA-appropriate priors. Our solution combines linear ICA with a dimensionality-preserving invertible flow f_θ . The flow acts between our data space of dimensionality and the representation fed to the linear ICA generative model; learning a representation that is well fit by the simple, linear ICA model. As we demonstrate in experiments (§5), this hybrid model, which we call *Bijecta*, succeeds where VAEs fail: it can match non-Gaussian priors and is able to discover independent latent sources on image datasets.

3.1 A Linear ICA base distribution for flows

Our aim here is to develop a non-square ICA method that is both end-to-end differentiable *and* computationally efficient, such that it can be trained jointly with a flow via stochastic gradient descent. We begin by choosing our base ICA source distribution to be a set of independent generalised Gaussian distributions, Eq (1) with $\mu = 0$, $\alpha = 1$ and ρ varying per experiment; and the ICA model’s likelihood to be a Gaussian.

$$p(s_i) = \text{GG}(s_i | \mu = 0, \alpha = 1, \rho), \text{ for } i \in \{1, \dots, d_s\},$$

$$p(\mathbf{z} | \mathbf{s}) = \mathcal{N}(\mathbf{x} | \mathbf{A}\mathbf{s}, \mathbf{\Sigma}_\theta),$$

where $\mathbf{A} \in \mathbb{R}^{d_x \times d_s}$ is our (unknown) ICA mixing matrix, which acts on the sources to produce a linear mixture; and $\mathbf{\Sigma}_\theta$ is a learnt or fixed diagonal covariance. This linear mixing of sources yields an intermediate representation \mathbf{z} that is then mapped to the data by a flow. Our model has three sets of variables: the observed data \mathbf{x} , the flow representation $\mathbf{z} = f^{-1}(\mathbf{x})$, and ICA latent sources \mathbf{s} .

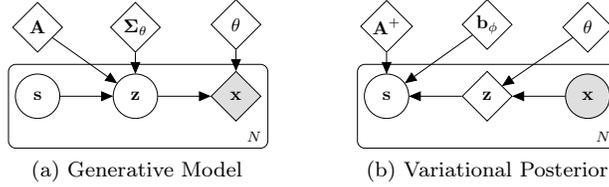


Figure 2: The generative model (a) and variational posterior (b), as defined in Eq (8).

It can be factorised as

$$p_\theta(\mathbf{x}, \mathbf{s}) = p_\theta(\mathbf{x}|\mathbf{s})p(\mathbf{s}) = p(\mathbf{z}|\mathbf{s})p(\mathbf{s}) \left| \det \frac{\partial f_\theta^{-1}}{\partial \mathbf{z}} \right| \quad (5)$$

While it is simple to train a flow by maximum likelihood method when we have a simple base distribution in \mathcal{Z} , here to obtain a maximum likelihood objective we would have to marginalise out \mathbf{s} to obtain the evidence in \mathcal{Z} ; a computationally intractable procedure:

$$p(\mathbf{z}; \mathbf{A}, \Sigma_\theta) = \int d\mathbf{s} p(\mathbf{z}|\mathbf{s}; \mathbf{A}, \Sigma_\theta) p(\mathbf{s}). \quad (6)$$

A contemporary approach is to use amortised variational inference for the linear ICA part of our model. This means we introduce an approximate amortised posterior for \mathbf{s} and perform importance sampling on Eq (6), taking gradients through our samples using the reparameterisation trick (Kingma & Welling, 2014; Rezende et al., 2014). Amortised stochastic variational inference offers numerous benefits: it scales training to large datasets by using stochastic gradient descent, our trained model can be applied to new data with a simple forward pass, and we are free to choose the functional & probabilistic form of our approximate posterior. Further our ICA model is end-to-end differentiable, making it optimal for jointly training with a flow.

We choose a linear mapping in our posterior, with $q_\phi(\mathbf{s}|\mathbf{z}) = \text{Laplace}(\mathbf{s}|\mathbf{A}^+\mathbf{z}, \mathbf{b}_\phi)$, where we have introduced variational parameters $\phi = \{\mathbf{A}^+, \mathbf{b}_\phi\}$ corresponding to an unmixing matrix and a diagonal diversity. Using samples from this posterior we can define a lower bound \mathcal{L} on the evidence in \mathcal{Z}

$$\begin{aligned} \log p(\mathbf{z}; \mathbf{A}, \Sigma_\theta) &\geq \mathcal{L}(\mathbf{z}; \phi, \mathbf{A}, \Sigma_\theta) \\ &= \mathbb{E}_{\mathbf{s} \sim q} [\log p(\mathbf{z}|\mathbf{s}) - \text{KL}(q_\phi(\mathbf{s}|\mathbf{z})||p(\mathbf{s}))] \end{aligned} \quad (7)$$

Using the change of variables equation, Eq (3), and the lower bound on the evidence for ICA in (7) for \mathcal{Z} , we can obtain a variational lower bound on the evidence for our data \mathbf{x} as the sum of the ICA model’s ELBO (acting on \mathbf{z}) and the log determinant of the flow:

$$\log p_\theta(\mathbf{x}; \mathbf{A}, \Sigma_\theta) \geq \mathcal{L}(\mathbf{x}; \theta, \phi, \mathbf{A}, \Sigma_\theta) = \mathcal{L}(\mathbf{z}; \phi, \mathbf{A}, \Sigma_\theta) + \log \left| \det \frac{\partial f_\theta^{-1}}{\partial \mathbf{z}} \right| \quad (8)$$

As such our model is akin to a flow model, but with an additional latent variable \mathbf{s} ; the base distribution $p(\mathbf{z})$ of the flow is defined through marginalizing out the linear mixing of the sources. We refer to a model with n non-linear splines mapping from \mathcal{X} to \mathcal{Z} as an n -layer Bijecta model.

In the case of non-square ICA, where our ICA model is not perfectly invertible, errors when reconstructing a mapping from \mathcal{S} to \mathcal{Z} may amplify when mapping back to \mathcal{X} . To mitigate this we add an additional regularisation term in our loss that penalises the L_1 error of each point when reconstructed into \mathcal{X} . This penalisation can be weighted according to the importance of high-fidelity reconstructions for a given application.

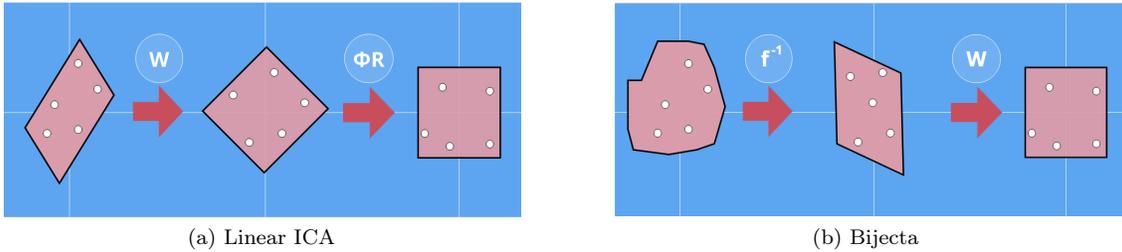


Figure 3: (a) Sequence of actions that are performed by the elements of \mathbf{A}^+ , the unmixing matrix of linear ICA. \mathbf{W} whitens the correlated data and $\Phi\mathbf{R}$ then ensures that the whitened (decorrelated) data is also independent. (b) Sequence of actions that are performed by the elements of Bijecta. f^{-1} maps data to a representation for which the whitening matrix *is* the ICA matrix. \mathbf{W} now whitens $f^{-1}(\mathbf{x})$ and the result is *also* statistically independent.

We attempted to train Bijecta with unconstrained mixing and unmixing matrices, but found that jointly training a linear model with a powerful flow was not trivial and models failed to converge when naively optimising Eq (8). We found it crucial to appropriately constrain the unmixing matrix to get models to converge. We detail these constraints in the next section.

4. Whitening in \mathbf{A}^+ , without SVD

What are good choices for the mixing and unmixing matrices? Recall in Sec 2.2 we discussed various traditional approaches to constraining the unmixing matrix. For our flow-based model, design choices as to the parameterisation of \mathbf{A}^+ stabilise and accelerate training. As before, the mixing matrix \mathbf{A} is unconstrained during optimisation. However, without the constraints on \mathbf{A}^+ we describe in this section, we found that joint training of a flow with linear ICA did not converge.

Recall Eq (2) — linear ICA methods carry out whitening \mathbf{W} , performing dimensionality reduction projecting from a d_x -dimensional space to a d_s -dimensional space, and the remaining rotation and scaling operations are square. When training with a flow the powerful splines we are learning can fulfill the role of the square matrices \mathbf{R} and Φ , but doing this ahead of the whitening itself. Put another way, the outputs from the flow can be learnt such that they are simply a whitening operation away from being effective ICA representations in \mathcal{S} . Thus, to minimise the complexities of jointly training a powerful flow with a small linear model, we can simply set $\mathbf{A}^+ = \mathbf{W}$, such that the unmixing matrix projects from d_z to d_s and is decorrelating. Statistical independence will come from the presence of the KL term in Eq (7): the flow will learn to give \mathbf{z} representations that, when whitened, are good ICA representations in \mathcal{S} . See Fig 3 for a visual illustration of this process and a comparison with the steps involved in linear ICA.

In previous linear ICA methods, the whitening procedure \mathbf{W} has been derived in some data-aware way. A common choice is to whiten via the Singular Value Decomposition (SVD) of the data matrix, where $\mathbf{W} = \Sigma\mathbf{U}^T$, Σ is the rectangular diagonal matrix of singular values of \mathbf{X} , and the columns of \mathbf{U} are the left-singular vectors. Computing the SVD of the whole dataset is expensive for large datasets; for us, in the context of Bijecta, we would be re-calculating the SVD of the representations $\mathbf{Z} = f^{-1}(\mathbf{X})$ of the entire dataset after every training step. One route around this would be online calculation of the whitening matrix (Cardoso & Laheld, 1996; Hyvärinen et al., 2001). This introduces an extra optimisation process that also has to be tuned, and would interact with the training of the flow.

To tackle these shortcomings of existing whitening methods, we propose a new method for linear non-square ICA that uses Johnson–Lindenstrauss (JL) transforms (also known as sketching) (Woodruff, 2014), which not only works effectively as a linear ICA method, but also works in conjunction with a flow model. These JL transforms have favourable properties for ICA, as we demonstrate in theoretical results. Further, this method samples part of the whitening matrix at initialisation and leaves it fixed for the remainder of training, requiring *no hyper-parameter tuning* and making it extremely computationally efficient. This method is novel and efficient when used as a whitening method within linear ICA, and when combined with a flow as in Bijeta is a powerful method for non-linear ICA as we demonstrate in experiments.

4.1 Approximately-Stiefel matrices

We have set $\mathbf{A}^+ = \mathbf{W}$, the whitening matrix. \mathbf{W} has two aims in non-square ICA. The first is dimensionality reduction, projecting from a d_x -dimensional space to a d_s -dimensional space. The second is to decorrelate the data it transforms, meaning that the resulting projection will have unit variance and mutually uncorrelated components. More formally we wish for \mathbf{W} of dimensionality $d_s \times d_x$ to be decorrelating.

The set of orthogonal decorrelating rectangular matrices lie on the Stiefel Manifold (Stiefel, 1935) denoted \mathcal{V} . For matrices with r rows and c columns, a matrix $\mathbf{G} \in \mathcal{V}(r, c)$ iff $\mathbf{G}\mathbf{G}^* = \mathbf{I}$ (\mathbf{G}^* the conjugate transpose of \mathbf{G}). Constraining the optimisation of \mathbf{W} to this manifold can be computationally expensive and complex (Bakir et al., 2004; Harandi & Fernando, 2016; Siegel, 2019) and instead we choose for \mathbf{W} to be *approximately* Stiefel, that is to lie close to $\mathcal{V}(d_s, d_x)$. This is justified by the following theorem, proved in Appendix A:

Theorem 1. *Let \mathbf{G} be a rectangular matrix and $\tilde{\mathbf{G}}$ be its projection onto $\mathcal{V}(r, c)$. As the Frobenius norm $\|\mathbf{G} - \tilde{\mathbf{G}}\| \rightarrow 0$ we have that $\|\mathbf{G}\mathbf{X}\mathbf{X}^T\mathbf{G}^T - \Psi\| \rightarrow 0$, where $\mathbf{G}\mathbf{X}\mathbf{X}^T\mathbf{G}^T$ is the cross-correlation of the projection of data \mathbf{X} by \mathbf{G} , and Ψ is some diagonal matrix.*

Simply put, this shows that as a matrix \mathbf{G} approaches the Stiefel manifold $\mathcal{V}(r, c)$ the off-diagonal elements of the cross-correlation matrix of the projection $\mathbf{G}\mathbf{X}$ are ever smaller, so \mathbf{G} is ever more decorrelating. Given these properties we want our whitening matrix to lie close to the Stiefel manifold.

4.1.1 JOHNSON-LINDENSTRAUSS PROJECTIONS

By Theorem 1 we know that we want our whitening matrix to be close to $\mathcal{V}(d_s, d_x)$. How might we enforce this closeness? By the definition of the Stiefel manifold, we can intuit that a matrix \mathbf{G} will lie close to this manifold if $\mathbf{G}\mathbf{G}^T \approx \mathbf{I}$. We formalise this as:

Theorem 2. *Let $\mathbf{G} \in \mathbb{R}^{d_s \times d_x}$ and let $\tilde{\mathbf{G}}$ be its projection onto $\mathcal{V}(d_s, d_x)$. As the Frobenius norm $\|\mathbf{G}\mathbf{G}^T - \mathbf{I}\| \rightarrow 0$, we also have $\|\tilde{\mathbf{G}} - \mathbf{G}\| \rightarrow 0$.*

The proof for this is presented in Appendix D. Using this theorem, we now propose an alternative to SVD-based whitening. Instead of having $\mathbf{W} = \Sigma^{-1}\mathbf{U}^T$ be the result of SVD on the data matrix, we define our whitening matrix as a data-independent Johnson–Lindenstrauss transform. We must ensure that \mathbf{W} , our rectangular matrix, is approximately orthogonal, lying close to the manifold $\mathcal{V}(d_s, d_x)$. More formally by Theorem 2, our goal is to construct a rectangular matrix \mathbf{W} such that $\mathbf{W}\mathbf{W}^T \approx \mathbf{I}$.

We construct approximately orthogonal matrices for \mathbf{W} by way of Johnson-Lindenstrauss (JL) Projections (Johnson & Lindenstrauss, 1984). A JL projection \mathbf{W} for $\mathbb{R}^{d_x} \rightarrow \mathbb{R}^{d_s}$ is sampled at

initialisation from a simple binary distribution Achlioptas (2003):

$$W_{i,j} = \begin{cases} +1/\sqrt{d_s}, & \text{with probability } \frac{1}{2} \\ -1/\sqrt{d_s}, & \text{with probability } \frac{1}{2} \end{cases} \quad (9)$$

This distribution satisfies $\mathbb{E}[\mathbf{W}\mathbf{W}^T] = \mathbf{I}$, and such a draw has $\mathbf{W}\mathbf{W}^T \approx \mathbf{I}$. We choose to fix \mathbf{W} after initialisation such that $\mathbf{A}^+ = \mathbf{W}$ never updates, greatly simplifying optimisation.

5. Experiments



Figure 4: Here we run linear ICA on a pair of images (a) that are mixed linearly (mix = $w_1 * \text{image}_1 + w_2 * \text{image}_2$) (b) to form a dataset with 512 points. In both cases w_1 and w_2 are sampled from a uniform distribution. We plot the mixing matrix \mathbf{A} for our JL-Cayley model with a quasi-uniform GG prior with $\rho = 10$ (c) and for FastICA (Hyvärinen & Oja, 1997) as a benchmark. \mathbf{A} should recover the source images, which occurs for both models.

Here we show that our approach outperforms VAEs and flows with ICA-priors at discovering ICA sources in image data. But first, as a sanity check, we show that a linear ICA model using JL projections to whiten can successfully unmix linearly mixed sources in Fig 4. For details on how to implement such a linear ICA model, see Appendix E. We take a pair of images from dSprites and create linear mixtures of them. We see that linear ICA with JL projections can successfully discover the true sources, the images used to create the mixtures, in the columns of \mathbf{A} .

Affine Data Given that we have established that our novel theory for decorrelating matrices can produce standalone linear ICA models, we now want to ascertain that our hybrid model performs well in non-linear mixing situations. To do so we create a dataset consisting of a subset of dSprites where we have a light-blue heart randomly uniformly placed on a black field. The true latent sources behind these randomly sampled affine transformations are simply the coordinates of the heart. First, in Fig B.2 we demonstrate that linear ICA models are unable to uncover the true latent sources. As expected non-linear mixing regimes motivate the use of flexible non-linear models.

We now demonstrate that Bijecta can uncover the latent sources underpinning these affine transformations, whereas VAEs with ICA-appropriate priors fail to do so. For details of VAE architecture, see Appendix I. These VAEs are able to learn to reconstruct data well, but the learnt latent space does not correspond to the underlying statistically independent sources (see Figs 1 and B.2). In fact for VAEs the effect of the latent variables is not consistent throughout the latent space, as seen in Fig 5. For Bijecta, the learnt latent space corresponds to the underlying statistically independent sources (see Figs 1 and B.2), and the meaning of the latent variables is consistent in Fig 5. Further in Fig 5 the model seems able to extrapolate outside the training domain: it generates images where the heart is partially rendered at the edges of the frame, even removing the heart entirely at times, even though such images are not in the training set.

Natural Images The previous experiments show that our model is capable of isolating independent sources on toy data. We complement this finding with experiments on a more complex natural

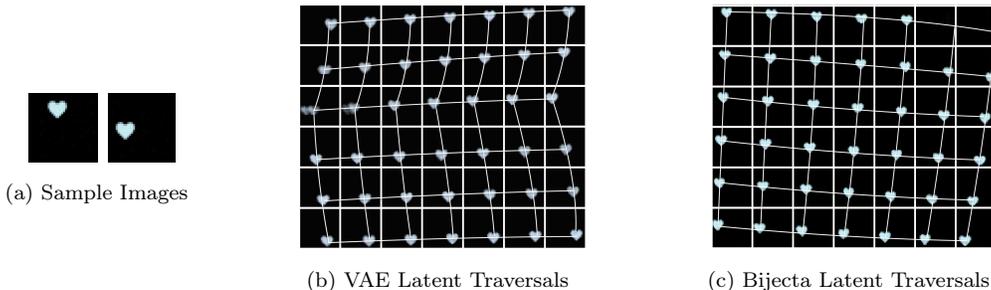


Figure 5: Here we demonstrate that Bijecta is capable of unmixing non-linearly mixed sources, better than VAEs with ICA-appropriate priors. We take a dSprites heart and, using a randomly sampled affine transformation, move it around a 32 by 32 background (a). With 2-D GG priors with $\rho = 10$ for a convolutional VAE (b) and for Bijecta (c) we plot the generations resulting from traversing the 2-D latent-source space in a square around the origin. We sketch the learnt axis of movement of the sprite with white lines. In (b) the VAE does not ascribe consistent meaning to its latent dimensions. It has failed to discover consistent independent latent sources: it has a sudden change in the learnt axes of movement along the second dimension, as seen by the kink in the white vertical lines. In (c) Bijecta is able to learn a simple affine transformation along each latent dimension, consistently spanning the space. In Fig B.2 we show the posterior distributions of both these models and show that Bijecta is better able to match the GG prior than the VAE, supporting our findings here.

image dataset, CelebA, and show that here too our model outperforms VAEs in learning factorisable representations.

An ersatz test of this can be done by synthesising images where we sample from a factorised approximation of Bijecta’s posterior. If the learned latent sources are actually independent, then the posterior over latent sources given the entire dataset should factorize into a product across dimensions, i.e. $q(\mathbf{s}) = \prod_i q(\mathbf{s}_i)$. In this case, we can fit an approximation to the posterior by fitting d_s independent one-dimensional density estimates on $q(\mathbf{s}_i)$. If the sources are not independent, then this factorized approximation to the posterior will be missing important correlations and dependencies. In Fig 6a samples from this factorised approximation look reasonable, suggesting that Bijecta has learnt representations that are statistically independent.

To quantify this source-separation numerically, we measure the total correlation (TC) of the aggregate posteriors of Bijecta ($q(\mathbf{s}|\mathbf{z})$) and VAEs ($q(\mathbf{z}|\mathbf{x})$) as Chen et al. (2018) do. Intuitively, the TC measures how well a distribution is approximated by the product of its marginals – or how much information is shared between variables due to dependence (Watanabe, 1960). It directly measures how well an ICA model has learnt decorrelated and independent latent representations (Everson & Roberts, 2001). Formally, it is the KL divergence between a distribution $r(\cdot)$ and a factorised representation of the distribution: $TC = \text{KL}(r(\mathbf{s}) || \prod_i r(s_i))$, where i indexes over (latent) dimensions.

In Table 1, we show that Bijecta learns an aggregate posterior with significantly lower TC values than both VAEs with Laplace priors, *and* β -TCVAEs – which in their training objective penalise the TC by a factor β (Chen et al., 2018). Our model has learnt a better ICA solution. We also include numerical results in Appendix J.1 showing that Bijecta outperforms linear ICA on a variety of natural image datasets.

Dimensionality reduction on flow models To conclude, having shown that Bijecta outperforms VAEs on a variety of non-linear ICA tasks, we now contrast our model’s ability to automatically uncover sources relative to flow models with heavy-tailed base distributions. We do so by measuring

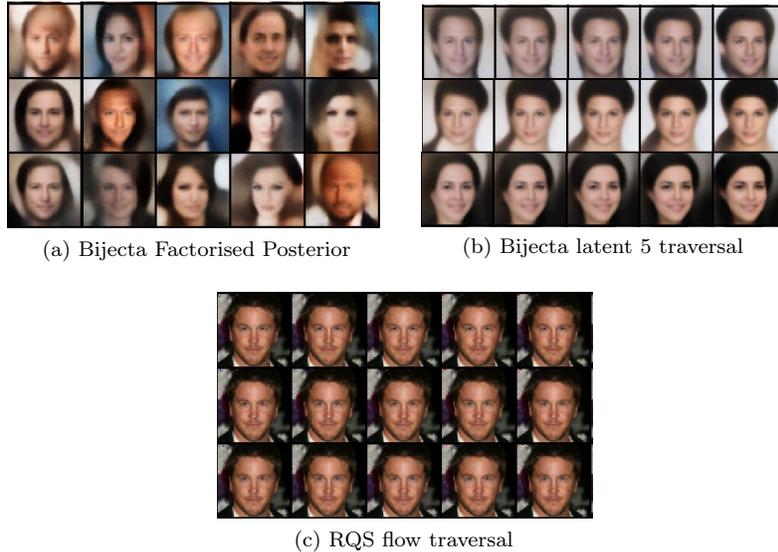


Figure 6: (a) shows decodings from an 8-layer Bijecta ($d_s = 32$) trained on CelebA with a Laplace prior (GG $\rho = 1$) where we sample from the factorised approximation to Bijecta’s posterior. See Fig J.6 for more such samples. (b) shows latent traversals for 3 different datapoints all along the same axis-aligned direction, for this same model. (c) shows traversals for a single embedded training datapoint from CelebA moving along 3 latent directions in an RQS flow with Laplace base distribution. Though we have selected 3 dimensions, all \mathcal{Z} dimensions had similar latent traversals. In (b-c) Images in the center correspond to the original latent space embedding, on either side we move up to 6 standard deviations away along this direction with other dimensions remaining fixed. The flow has not discovered axis-aligned transforms, whereas Bijecta has learned informative latent dimensions: here the dimension encodes hair thickness. Note that identity is maintained throughout and that the transform is consistent across different posterior samples. See Appendix J.3 for gallery of transforms for Bijecta.

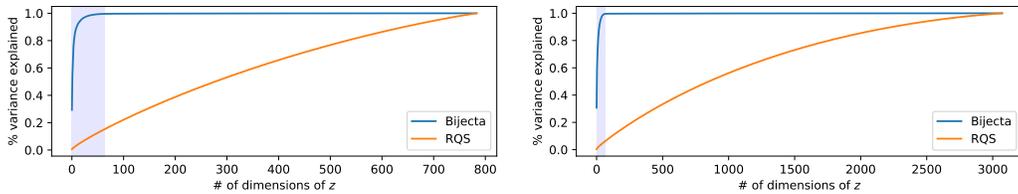


Figure 7: Explained variance plots for the embedding in \mathcal{Z} , as measured by the sums of the eigenvalues of the covariance matrix of the embeddings, for both our Bijecta model and for an RQS model of equivalent size trained with a Laplace base distribution (GG distribution with $\rho = 1$). For both Fashion-MNIST (left) and CIFAR 10 (right) datasets we see that the Bijecta model has learned a compressive flow, where most of the variance can be explained by only a few linear projections. The shaded region denotes the first 64 dimensions, corresponding to the size of the target source embedding \mathcal{S} .

the cumulative explained variance by the dimensions in \mathcal{Z} for both models. If a small number of dimensions explains most of \mathcal{Z} ’s variance then the model has learnt a bijection which only requires a

Table 1: Total Correlation Results: We evaluate the source separation of different models on CelebA via the TC of the validation set embeddings in the 32-D latent space of: Laplace prior VAEs, β -TCVAEs ($\beta = 15$), and Bijecta with a Laplace prior (\pm indicates the standard deviation over 2 runs). VAEs use the same architecture and training as Chen et al. (2018).

	Laplace-VAE	β -TCVAE	Laplace-Bijecta
TC:	106.7 ± 0.9	55.7 ± 0.1	13.1 ± 0.4

small number of dimensions to be invertible. It has in effect learnt the generating sources underpinning the data.

In Fig 7 we show that Bijecta induces better-compressed representations in \mathcal{Z} than non-compressive flows. We plot the eigenvalues of the covariance matrix on the output of the flow, i.e. on $\text{Cov}(f(\mathbf{X}))$, to see how much of the total variance in the learned feature space \mathcal{Z} can be explained in a few dimensions. In doing so we see that a flow trained jointly with a linear ICA model with $d_s = 64$ effectively concentrates variation into a small number of intrinsic dimensions; this is in stark contrast with the RQS flows trained with only a Laplace base distribution. This demonstrates that our model is able to automatically detect relevant directions on a low dimensional manifold in \mathcal{Z} , and that the bijective component of our model is better able to isolate latent sources than a standard flow.

For a visual illustration of this source separation we show the difference in generated images resulting from smoothly varying along each dimension in \mathcal{S} for Bijecta models and in \mathcal{Z} for flows in Fig 6. Bijecta is clearly able to discover latent sources, whereby it learns axis-aligned transformations of CelebA faces, whereas a flow with equivalent computational budget and a heavy-tailed base distribution is not able to.

All flow-based baselines are trained using the objective in Eq (4), using Real-NVP style factoring-out (Durkan et al., 2019; Dinh et al., 2015), and are matched in size and neural network architectures to the flows of Bijecta models. See Appendix I for more details.

6. Related Work

One approach to extend ICA to non-linear settings is to have a non-linear mapping acting on the independent sources and data (Burel, 1992; Deco & Brauer, 1995; Yang et al., 1998; Valpola et al., 2003). In general, non-linear ICA models have been shown to be hard to train, having problems of unidentifiability: the model has numerous local minima it can reach under its training objective, each with potentially different learnt sources (Hyvärinen & Pajunen, 1999; Karhunen, 2001; Almeida, 2003; Hyvarinen et al., 2019). Some non-linear ICA models have been specified with additional structure to reduce the space of potential solutions, such as putting priors on variables (Lappalainen & Honkela, 2000) or specifying the precise non-linear functions involved (Lee & Koehler, 1997; Taleb, 2002). Recent work shows that conditioning the source distributions on some always-observed side information, say time index, can be sufficient to induce identifiability in non-linear ICA (Khemakhem et al., 2020).

Modern flows were first proposed as an approach to non-linear square ICA (Dinh et al., 2015), but are also motivated by desires for more expressive priors and posteriors (Kingma et al., 2016; Papamakarios et al., 2019). Early approaches, known as symplectic maps (Deco & Brauer, 1995; Parra et al., 1995, 1996), were also proposed for use with ICA. Flows offer expressive dimensionality-preserving (and sometimes volume-preserving) bijective mappings (Dinh et al., 2017; Kingma & Dhariwal, 2018). Flows have been used to provide feature extraction for linear discriminative models (Nalisnick et al., 2019). Orthogonal transforms have been used in normalizing flows before, to improve the optimisation

properties of Sylvester flows (Van Den Berg et al., 2018; Golinski et al., 2019). Researchers have also looked at constraining neural network weights to the Stiefel-manifold (Li et al., 2020).

7. Conclusion

We have developed a method for performing non-linear ICA large high-dimensional image datasets which combines state-of-the-art flow-based models and a novel theoretically grounded linear ICA method. This model succeeds where existing probabilistic deep generative models fail: its constituent flow is able to learn a representation, lying in a low dimensional manifold in \mathcal{Z} , under which sources are separable by linear unmixing. In source space \mathcal{S} , this model learns a low dimensional, explanatory set of statistically independent latent sources.

References

- Absil, P. A. & Malick, J. (2012). Projection-like retractions on matrix manifolds. *SIAM Journal on Optimization*, 22(1), 135–158.
- Achlioptas, D. (2003). Database-friendly random projections: Johnson-Lindenstrauss with binary coins. In *Journal of Computer and System Sciences*, volume 66 (pp. 671–687).
- Almeida, L. B. (2003). MISEP – Linear and Nonlinear ICA Based on Mutual Information. *Journal of Machine Learning Research*, 4, 1297–1318.
- Bakir, G. H., Gretton, A., Franz, M., & Schölkopf, B. (2004). Multivariate regression via Stiefel manifold constraints. In *Joint Pattern Recognition Symposium* (pp. 262–269).
- Bell, A. J. & Sejnowski, T. J. (1995). An information-maximisation approach to blind separation and blind deconvolution. *Neural Computation*, 7(6), 1004–1034.
- Burel, G. (1992). Blind separation of sources: A nonlinear neural algorithm. *Neural Networks*, 5(6), 937–947.
- Cardoso, J. F. (1989a). Blind identification of independent components with higher-order statistics. In *IEEE Workshop on Higher-Order Spectral Analysis*.
- Cardoso, J. F. (1989b). Source separation using higher order moments. In *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, volume 4 (pp. 2109–2112).
- Cardoso, J. F. (1997). Infomax and Maximum Likelihood for Blind Source Separation. *IEEE Letters on Signal Processing*, 4, 112–114.
- Cardoso, J. F. & Laheld, B. H. (1996). Equivariant adaptive source separation. *IEEE Transactions on Signal Processing*, 44(12), 3017–3030.
- Cayley, A. (1846). Sur quelques propriétés des déterminants gauches. *Journal für die reine und angewandte Mathematik*, 32, 119–123.
- Chen, R. T. Q., Li, X., Grosse, R., & Duvenaud, D. (2018). Isolating Sources of Disentanglement in Variational Autoencoders. In *NeurIPS*.
- Choudrey, R. (2000). *Variational Methods for Bayesian Independent Component Analysis*. PhD thesis, University of Oxford.
- Comon, P. (1994). Independent component analysis, A new concept? *Signal Processing*, 36(3), 287–314.
- Dasgupta, S. & Gupta, A. (2003). An Elementary Proof of a Theorem of Johnson and Lindenstrauss. *Random Structures and Algorithms*, 22(1), 60–65.
- Deco, G. & Brauer, W. (1995). Higher Order Statistical Decorrelation without Information Loss. In *NeurIPS*.
- Dinh, L., Krueger, D., & Bengio, Y. (2015). NICE: Non-linear Independent Components Estimation. In *ICLR*.
- Dinh, L., Sohl-Dickstein, J., & Bengio, S. (2017). Density estimation using Real NVP. In *ICLR*.
- Durkan, C., Bekasov, A., Murray, I., & Papamakarios, G. (2019). Neural Spline Flows. In *NeurIPS*.
- Everson, R. & Roberts, S. J. (1999). Independent Component Analysis: A Flexible Nonlinearity and Decorrelating Manifold Approach. *Neural Computation*, 11(8), 1957–83.

- Everson, R. & Roberts, S. J. (2001). *Independent Component Analysis*. Cambridge University Press.
- Golinski, A., Rainforth, T., & Lezcano-Casado, M. (2019). Improving Normalizing Flows via Better Orthogonal Parameterizations. In *ICML Workshop on Invertible Neural Networks and Normalizing Flows*.
- Harandi, M. & Fernando, B. (2016). Generalized BackPropagation, Etude De Cas: Orthogonality.
- Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., & Lerchner, A. (2017). β -VAE: Learning Basic Visual Concepts with a Constrained Variational Framework. In *ICLR*.
- Hyvärinen, A., Karhunen, J., & Oja, E. (2001). *Independent Component Analysis*. John Wiley.
- Hyvärinen, A. & Oja, E. (1997). A fast fixed-point algorithm for independent component analysis. *Neural Computation*, 9(7), 1483–1492.
- Hyvärinen, A. & Pajunen, P. (1999). Nonlinear independent component analysis: Existence and uniqueness results. *Neural Networks*, 12(3), 429–439.
- Hyvarinen, A., Sasaki, H., & Turner, R. E. (2019). Nonlinear ICA Using Auxiliary Variables and Generalized Contrastive Learning. In *AISTATS*.
- Johnson, W. B. & Lindenstrauss, J. (1984). Extensions of Lipschitz mappings into a Hilbert space. *Contemporary mathematics*, 26(1), 189–206.
- Karhunen, J. (2001). Nonlinear Independent Component Analysis. In R. Everson & S. J. Roberts (Eds.), *ICA: Principles and Practice* (pp. 113–134). Cambridge University Press.
- Khemakhem, I., Kingma, D. P., Monti, R. P., & Hyvärinen, A. (2020). Variational Autoencoders and Nonlinear ICA: A Unifying Framework. In *AISTATS*.
- Kingma, D. P. & Dhariwal, P. (2018). Glow: Generative flow with invertible 1x1 convolutions. *NeurIPS*.
- Kingma, D. P. & Lei Ba, J. (2015). Adam: A Method for Stochastic Optimisation. In *ICLR*.
- Kingma, D. P., Salimans, T., Jozefowicz, R., Chen, X., Sutskever, I., & Welling, M. (2016). Improved Variational Inference with Inverse Autoregressive Flow. In *NeurIPS*.
- Kingma, D. P. & Welling, M. (2014). Auto-encoding Variational Bayes. In *ICLR*.
- Lappalainen, H. & Honkela, A. (2000). Bayesian Non-Linear Independent Component Analysis by Multi-Layer Perceptrons. In M. Girolami (Ed.), *Advances in Independent Component Analysis* (pp. 93–121). Springer.
- Lawrence, N. D. & Bishop, C. M. (2000). *Variational Bayesian Independent Component Analysis*. Technical report, University of Cambridge.
- Lee, T.-W., Girolami, N., Bell, A. J., & Sejnowski, T. J. (2000). A Unifying Information-Theoretic Framework for Independent Component Analysis. *Computers & Mathematics with Applications*, 39(11), 1–21.
- Lee, T. W. & Koehler, B. U. (1997). Blind source separation of nonlinear mixing models. *Neural Networks for Signal Processing - Proceedings of the IEEE Workshop*, (pp. 406–415).
- Lee, T.-W. & Sejnowski, T. J. (1997). Independent Component Analysis for Mixed Sub-Gaussian and Super-Gaussian Sources. *Joint Symposium on Neural Computation*, (pp. 6–13).

- Li, J., Li, F., & Todorovic, S. (2020). Efficient Riemannian Optimization on the Stiefel Manifold via the Cayley Transform. In *ICLR*.
- Locatello, F., Bauer, S., Lucie, M., Rätsch, G., Gelly, S., Schölkopf, B., & Bachem, O. (2019). Challenging common assumptions in the unsupervised learning of disentangled representations. In *ICML*, volume 2019-June.
- Mackay, D. J. C. (1996). *Maximum Likelihood and Covariant Algorithms for Independent Component Analysis*. Technical report, University of Cambridge.
- Mathieu, E., Rainforth, T., Siddharth, N., & Teh, Y. W. (2019). Disentangling Disentanglement in Variational Autoencoders. In *ICML*.
- Nalisnick, E., Matsukawa, A., Teh, Y. W., & Lakshminarayanan, B. (2019). Detecting Out-of-Distribution Inputs to Deep Generative Models Using Typicality.
- Papamakarios, G., Nalisnick, E., Rezende, D. J., Mohamed, S., & Lakshminarayanan, B. (2019). *Normalizing Flows for Probabilistic Modeling and Inference*. Technical report, DeepMind, London, UK.
- Parra, L., Deco, G., & Miesbach, S. (1995). Redundancy reduction with information-preserving nonlinear maps. *Network: Computation in Neural Systems*, 6(1), 61–72.
- Parra, L., Deco, G., & Miesbach, S. (1996). Statistical Independence and Novelty Detection with Information Preserving Nonlinear Maps. *Neural Computation*, 8(2), 260–269.
- Rezende, D. J. & Mohamed, S. (2015). Variational Inference with Normalizing Flows. In *ICML*.
- Rezende, D. J., Mohamed, S., & Wierstra, D. (2014). Stochastic Backpropagation and Approximate Inference in Deep Generative Models. In *ICML*.
- Rolinek, M., Zietlow, D., & Martius, G. (2019). Variational autoencoders pursue pca directions (by accident). In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2019-June (pp. 12398–12407).
- Roweis, S. & Ghahramani, Z. (1999). A unifying review of linear gaussian models. *Neural Computation*, 11(2), 305–345.
- Siegel, J. W. (2019). Accelerated Optimization With Orthogonality Constraints.
- Stiefel, E. (1935). Richtungsfelder und Fernparallelismus in n-dimensionalen Mannigfaltigkeiten. *Commentarii mathematici Helvetici*, 8, 305–353.
- Stühmer, J., Turner, R. E., & Nowozin, S. (2019). Independent Subspace Analysis for Unsupervised Learning of Disentangled Representations.
- Taleb, A. (2002). A generic framework for blind source separation in structured nonlinear models. *IEEE Transactions on Signal Processing*, 50(8), 1819–1830.
- Valpola, H., Oja, E., Ilin, A., Honkela, A., & Karhunen, J. (2003). Nonlinear blind source separation by variational Bayesian learning. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, E86-A(3), 532–541.
- Van Den Berg, R., Hasenclever, L., Tomczak, J. M., & Welling, M. (2018). Sylvester normalizing flows for variational inference. In *UAI*, volume 1 (pp. 393–402).
- Watanabe, S. (1960). Information Theoretical Analysis of Multivariate Correlation. *IBM Journal of Research and Development*, 4(1), 66–82.

- Woodruff, D. P. (2014). Sketching as a Tool for Numerical Linear Algebra. *Foundations and Trends in Theoretical Computer Science*, 10(2), 1–157.
- Yang, H. H., Amari, S. I., & Cichocki, A. (1998). Information-theoretic approach to blind separation of sources in non-linear mixture. *Signal Processing*, 64(3), 291–300.

Appendix for Learning Bijective Feature Maps for Linear ICA

Appendix A. Correlated and Dependent Sources

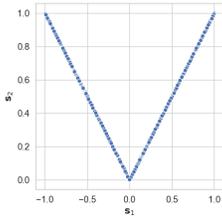


Figure A.1: Sources can be uncorrelated *and* dependent. Consider our first source \mathbf{s}_1 to be uniformly distributed on the interval $[-1, 1]$. If $\mathbf{s}_1 \leq 0$, then $\mathbf{s}_2 = -\mathbf{s}_1$, else $\mathbf{s}_2 = \mathbf{s}_1$. In this case the variables are uncorrelated, $\mathbb{E}[\mathbf{s}_1\mathbf{s}_2] = 0$, but the joint distribution of \mathbf{s}_1 and \mathbf{s}_2 is not uniform on the rectangle $[-1, 1] \times [0, 1]$, as it would be if they were independent. See plot to the left for an illustration of this.

Appendix B. VAEs underperform on ICA

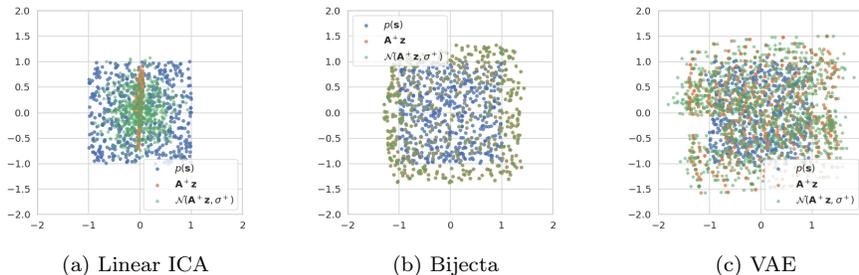


Figure B.2: In (a), (b) we run linear ICA and a single-layer Bijecta on the affine transformation dataset of Fig 5. We take a dSprites heart and using a randomly sampled affine transformation, move it around a 32 by 32 background. We plot the posterior distribution $\mathcal{N}(\mathbf{A}^+\mathbf{z}, \sigma^+)$ (green) and its mean $\mathbf{A}^+\mathbf{z}$ (orange) for both models. Clearly the posterior from Bijecta is better able to match the quasi-uniform GG prior with $\rho = 10$ ($p(\mathbf{s})$ in blue) than the linear ICA model, highlighting that the addition of the flow allows for linear unmixing. Similarly the large VAE of Fig 5 is unable to match this prior, forming two posterior ‘lobes’ around the prior.

Appendix C. Proof of optimality

Definition: We say a matrix \mathbf{B}' is strictly more orthogonal than a matrix \mathbf{B} if $\|\mathbf{B}'^T \mathbf{B}' - \mathbf{I}\| < \|\mathbf{B}^T \mathbf{B} - \mathbf{I}\|$.

Theorem 1. As the Frobenius norm $\|\mathbf{G} - \tilde{\mathbf{G}}\| \rightarrow 0$, where $\mathbf{G} \in \mathbb{R}^{r \times c}$ and $\tilde{\mathbf{G}}$ is the projection of \mathbf{G} onto $\mathcal{V}(r, c)$, $\|\mathbf{G} \mathbf{X} \mathbf{X}^T \mathbf{G}^T - \mathbf{D}\| \rightarrow 0$, where $\mathbf{G} \mathbf{X} \mathbf{X}^T \mathbf{G}^T$ is the cross-correlation of the projection of data \mathbf{X} by \mathbf{G} , and \mathbf{D} is some diagonal matrix.

Proof. The Stiefel manifold is given by (assuming $r < c$):

$$\mathcal{V}(r, c) = \{\mathbf{G} \in \mathbb{R}^{r \times c} : \mathbf{G} \mathbf{G}^T = \mathbf{I}\} \quad (1)$$

The unique projection $\tilde{\mathbf{G}}$ onto this manifold of a matrix $\mathbf{G} \in \mathbb{R}^{d_s \times d_x}$, with polar decomposition $\mathbf{U} \mathbf{P}$, is simply \mathbf{U} (Absil & Malick, 2012). \mathbf{P} denotes a $c \times c$ positive-semidefinite Hermitian matrix and \mathbf{U} is a $r \times c$ orthogonal matrix, i.e $\mathbf{U} \in \mathcal{V}(r, c)$ and has a conjugate transpose denoted \mathbf{U}^* such that $\mathbf{U} \mathbf{U}^* = \mathbf{I}$. As such given any matrix \mathbf{G} we have a polar decomposition $\mathbf{G} = \tilde{\mathbf{G}} \mathbf{P}$, where $\tilde{\mathbf{G}}$, a linear decorrelating matrix, is the projection onto $\mathcal{V}(r, c)$ and \mathbf{P} denotes a $r \times c$ positive-semidefinite Hermitian matrix.

Let $\mathbf{G} \in \mathbb{R}^{r \times c}$ be some projection matrix of data \mathbf{X} . We have $\mathbf{S} = \mathbf{G} \mathbf{X}$. The cross-correlation is expressed as $\mathbf{S} \mathbf{S}^T = \mathbf{G} \mathbf{X} \mathbf{X}^T \mathbf{G}^T$. In the case where \mathbf{S} is perfectly decorrelated, we have: $\mathbf{S} \mathbf{S}^T = \mathbf{D}^2$ where \mathbf{D} is a diagonal matrix. We know that the Stiefel manifold $\mathcal{V}(r, c)$ (defined in Eq (1)) holds the set of all whitening matrices (Everson & Roberts, 1999), up to the diagonal matrix \mathbf{D}^{-1} .

For any matrix \mathbf{G} and its projection $\tilde{\mathbf{G}}$ onto $\mathcal{V}(r, c)$, we have $\tilde{\mathbf{G}} \mathbf{X} \mathbf{X}^T \tilde{\mathbf{G}}^* = \mathbf{D}^2$, where $\tilde{\mathbf{G}}^*$ is the complex conjugate of $\tilde{\mathbf{G}}$. Consequently, given that the Frobenius norm is unitary invariant and the fact that $\tilde{\mathbf{G}}$ is unitary:

$$\begin{aligned} \|\mathbf{G} \mathbf{X} \mathbf{X}^T \mathbf{G}^T - \mathbf{D}^2\| &= \|\tilde{\mathbf{G}} \mathbf{P} \mathbf{X} \mathbf{X}^T \mathbf{P}^T \tilde{\mathbf{G}}^* - \mathbf{D}^2\| \\ &= \|\tilde{\mathbf{G}} \mathbf{P} \mathbf{X} \mathbf{X}^T \mathbf{P}^T \tilde{\mathbf{G}}^* - \tilde{\mathbf{G}} \mathbf{X} \mathbf{X}^T \tilde{\mathbf{G}}^*\| \\ &= \|\mathbf{P} \mathbf{X} \mathbf{X}^T \mathbf{P}^T - \mathbf{X} \mathbf{X}^T\| \\ &\leq \|\mathbf{P}^2 - \mathbf{I}\| \|\mathbf{X} \mathbf{X}^T\| \end{aligned}$$

The last line comes from the triangle inequality and that \mathbf{P} is Hermitian so $\mathbf{P} \mathbf{P}^T = \mathbf{P}^2$. As $\|\mathbf{X} \mathbf{X}^T\|$ is a constant, $\|\mathbf{P}^2 - \mathbf{I}\| \|\mathbf{X} \mathbf{X}^T\| \rightarrow 0$ as $\|\mathbf{P}^2 - \mathbf{I}\| \rightarrow 0$.

Proposition 1. Let $\mathbf{G} \in \mathbb{R}^{r \times c}$ and $\tilde{\mathbf{G}}$ is the projection of \mathbf{G} onto $\mathcal{V}(r, c)$. Further let $\mathbf{G} = \tilde{\mathbf{G}} \mathbf{P}$. Then $\|\mathbf{G} - \tilde{\mathbf{G}}\| \rightarrow 0 \Leftrightarrow \|\mathbf{P}^2 - \mathbf{I}\| \rightarrow 0$.

Proof. As the Frobenius norm is invariant to unitary transformations, and because $\tilde{\mathbf{G}}$ and \mathbf{U} are unitary matrices, the norm between $\tilde{\mathbf{G}}$ and \mathbf{G} is

$$\begin{aligned} \|\mathbf{G} - \tilde{\mathbf{G}}\| &= \|\mathbf{G} \tilde{\mathbf{G}}^* - \mathbf{I}\| \\ &= \|\mathbf{U} \mathbf{P} \mathbf{U}^* - \mathbf{I}\| \\ &= \|\mathbf{U} (\mathbf{P} - \mathbf{I}) \mathbf{U}^*\| \\ &= \|\mathbf{P} - \mathbf{I}\|. \end{aligned}$$

\mathbf{P} is positive-semidefinite and hence $\|\mathbf{P}^2 - \mathbf{I}\| \rightarrow 0$ implies $\|\mathbf{P} - \mathbf{I}\| \rightarrow 0$ and the distance between \mathbf{G} and $\mathcal{V}(|\mathbf{s}|, |\mathbf{z}|)$ strictly decreases, i.e $\|\mathbf{G} - \tilde{\mathbf{G}}\| \rightarrow 0$. Generally:

$$\|\mathbf{P}^2 - \mathbf{I}\| \rightarrow 0 \Leftrightarrow \|\mathbf{G} - \tilde{\mathbf{G}}\| \rightarrow 0 \quad (2)$$

This ends the proof. \square

Thus by Proposition 1 we have that:

$$\|\mathbf{G} - \tilde{\mathbf{G}}\| \rightarrow 0 \Leftrightarrow \|\mathbf{G}\mathbf{X}\mathbf{X}^T\mathbf{G}^T - \mathbf{D}\| \rightarrow 0. \quad (3)$$

This ends the proof. \square

Appendix D. Proof of closeness

Theorem 2. Let $\mathbf{G} \in \mathbb{R}^{d_s, D}$ and let $\tilde{\mathbf{G}}$ be its projection onto $\mathcal{V}(d_s, d_x)$. As the Frobenius norm $\|\mathbf{G}\mathbf{G}^T - \mathbf{I}\| \rightarrow 0$, we also have $\|\tilde{\mathbf{G}} - \mathbf{G}\| \rightarrow 0$.

Proof. Let $\mathbf{G} \in \mathbb{R}^{d_s, D}$ and let $\tilde{\mathbf{G}}$ be its projection onto $\mathcal{V}(d_s, d_x)$. By Proposition 1 in Appendix C we have that

$$\|\mathbf{P}^2 - \mathbf{I}\| \rightarrow 0 \Leftrightarrow \|\mathbf{G} - \tilde{\mathbf{G}}\| \rightarrow 0$$

$\mathbf{G} \in \mathbb{R}^{d_s \times d_x}$, has polar decomposition $\mathbf{U}\mathbf{P}$. Note that $\mathbf{G}\mathbf{G}^* = \mathbf{U}\mathbf{P}\mathbf{P}^T\mathbf{U}^* = \mathbf{U}\mathbf{P}^2\mathbf{U}^*$ because \mathbf{P} is Hermitian. Recall that the Frobenius norm is invariant to unitary transformations, such as \mathbf{U} :

$$\begin{aligned} \|\mathbf{G}\mathbf{G}^* - \mathbf{I}\| &= \|\mathbf{U}\mathbf{P}^2\mathbf{U}^* - \mathbf{I}\| \\ &= \|\mathbf{P}^2 - \mathbf{I}\|. \end{aligned}$$

As such:

$$\|\mathbf{G}\mathbf{G}^* - \mathbf{I}\| \rightarrow 0 \Leftrightarrow \|\mathbf{G} - \tilde{\mathbf{G}}\| \rightarrow 0 \quad (4)$$

Note that we can trivially show this for $\mathbf{G}^*\mathbf{G} = \mathbf{P}^2$.

This ends the proof. \square

Appendix E. Linear ICA using JL Projections

Instead of having $\mathbf{W} = \Sigma^{-1}\mathbf{U}^T$ be the result of SVD on the data matrix, we define our whitening matrix as $\mathbf{W} = \mathbf{\Lambda}\mathbf{Q}$, a learnt diagonal matrix $\mathbf{\Lambda} \in \mathbb{R}^{d_s}$ and a fixed, data-independent rectangular matrix $\mathbf{Q} \in \mathbb{R}^{d_s \times d_x}$. We must ensure that \mathbf{Q} , our rectangular matrix, is approximately orthogonal, lying close to the manifold $\mathcal{V}(d_s, d_x)$.

As with the ICA model for Bijecta we construct approximately orthogonal matrices for \mathbf{Q} by way of Johnson-Lindenstrauss (JL) Projections (Johnson & Lindenstrauss, 1984; Dasgupta & Gupta, 2003). A JL projection \mathbf{Q} for $\mathbb{R}^{d_x} \rightarrow \mathbb{R}^{d_s}$ is sampled at initialisation from a simple binary distribution with equal probability (Achlioptas, 2003) such that:

$$Q_{i,j} = \pm 1/\sqrt{d_s}. \quad (5)$$

This distribution satisfies $\mathbb{E}[\mathbf{Q}\mathbf{Q}^T] = \mathbf{I}$, and, as shown (Achlioptas, 2003), such a draw has $\mathbf{Q}\mathbf{Q}^T \approx \mathbf{I}$. This gives us our factorised form for \mathbf{A}^+ ,

$$\mathbf{A}^+ = \Phi\mathbf{R}\Lambda\mathbf{Q}. \quad (6)$$

¹.

We can choose to fix \mathbf{Q} after initialisation such that optimisation occurs in the significantly smaller $\mathbb{R}^{\frac{1}{2}d_s \times (d_s+3)}$ space, solely for the matrix $\Phi\mathbf{R}\Lambda$, greatly simplifying our optimisation problem. We draw \mathbf{Q} at initialisation and leave it fixed for the remainder of optimisation, making it a much more efficient method than SVD-based whitening.

By Theorems 1 and 2 we know that \mathbf{A}^+ using \mathbf{Q} as a component will be close to the manifold of decorrelating matrices, Eq (6). In the next section, we detail constraints on the only matrix that is optimised, $\Phi\mathbf{R}$, which improve convergence.

E.1 The $SO(d_s)$ Lie group for \mathbf{R}

Recall that \mathbf{R} is a square and orthogonal decorrelating matrix of dimensionality $\mathbb{R}^{d_s \times d_s}$. As such, we can constrain \mathbf{R} to be in the orthogonal group $O(d_s)$, which has been shown to be optimal for decorrelating ICA sources. In the case of square ICA, with sufficient data, the maximum likelihood unmixing matrices are members of this group and will be reached by models confined to this group (Everson & Roberts, 1999).

We want to perform unconstrained optimisation in learning our matrix, so we wish to use a differentiable transformation from a class of simpler matrices to $O(d_s)$. One such transform is the Cayley transform (Cayley, 1846), which maps a given anti-symmetric matrix \mathbf{M} (i.e., satisfying $\mathbf{M} = -\mathbf{M}^T$) to the group of special orthogonal matrices $SO(d_s)$ with determinant 1. $SO(d_s)$ is the elements of the group $O(d_s)$ with determinant 1. Unlike $O(d_s)$ it is path-connected, aiding optimisation. As such, we propose defining our square unmixing matrix using the Cayley transform of the anti-symmetric matrix \mathbf{M} ,

$$\mathbf{R} = (\mathbf{I} - \mathbf{M})^{-1}(\mathbf{I} + \mathbf{M}). \quad (7)$$

This can be formulated as an unconstrained problem, easing optimisation, by defining $\mathbf{M} = (\mathbf{L} - \mathbf{L}^T)/2$ and then optimising over the square real-valued matrix \mathbf{L} . This further reduces the optimisation space for \mathbf{A}^+ to $\mathbb{R}^{\frac{1}{2}d_s(d_s+3)}$.

Appendix F. Sub- or Super-Gaussian Sources?

In classical non-linear ICA a simple non-linear function (such as a matrix mapping followed by an activation function) is used to map directly from data to the setting of the sources for that datapoint (Bell & Sejnowski, 1995). In this noiseless model, the activation function is related to the prior one is implicitly placing over the sources (Roweis & Ghahramani, 1999). The choice of non-linearity here is thus a claim on whether the sources were sub- or super- Gaussian. If the choice is wrong, ICA models struggle to unmix the data (Roweis & Ghahramani, 1999). Previous linear ICA methods enabled both mixed learning of super and sub sources (Lee & Sejnowski, 1997), and learning the nature of the source (Everson & Roberts, 1999).

Appendix G. Identifiability of Linear ICA

Recall that for noiseless linear ICA the learnt sources will vary between different trained models only in their ordering and scaling Choudrey (2000); Hyvärinen et al. (2001); Everson & Roberts (2001).

1. We use capitalised bold Greek letters for diagonal matrices, capitalised bold Roman letters otherwise

Under this model our data matrix \mathbf{X} is

$$\mathbf{X} = \mathbf{A}\mathbf{S}, \tag{8}$$

where each column of \mathbf{S} is distributed according to $p(\mathbf{s})$. Given a permutation matrix \mathbf{P} and a diagonal matrix \mathbf{D} , both $\in \mathbb{R}^{d_s \times d_s}$, we define new source and mixing matrices such that \mathbf{X} is unchanged: $\mathbf{A} \leftarrow \mathbf{A}\mathbf{P}\mathbf{D}$ and $\mathbf{S} \leftarrow \mathbf{P}\mathbf{D}^{-1}\mathbf{S}$ Choudrey (2000). With fixed source variance the scaling ambiguity is removed, so linear ICA is easy to render identifiable up to a permutation and sign-flip of sources. However, this is only the case when the underlying sources are non-Gaussian (Hyvärinen et al., 2001). The Bijecta model can be interpreted as learning the “data” for linear ICA, with

$$f(\mathbf{X}) \approx \mathbf{A}\mathbf{S}. \tag{9}$$

In this setting, a natural question to ask is whether or not, given a particular function $f : \mathcal{X} \rightarrow \mathcal{Z}$, the linear ICA is identifiable. The Bijecta model explicitly aims to induce this linear identifiability on its feature map, as we impose a Laplace prior $p(\mathbf{s})$, with fatter-than-Gaussian tails.

Appendix H. Coupling Layers in Flows

Coupling layers in flows are designed to produce lower triangular Jacobians for ease of calculation of determinants. RQS-flows are defined by K monotonically increasing *knots*, which are the coordinate pairs through which the function passes: $\{(x_k, y_k)\}_{k=0}^K$. We can interpolate values between each of the K knots using the equation for the RQS transformation Durkan et al. (2019). The resulting function is a highly flexible non-linear transformation, such that RQS flows require fewer composed mappings to achieve good performance relative to other coupling layers. The knots themselves are trainable, and parameterised by deep neural networks. These can then be composed with other tractable transformations, including permutations and multiplication by triangular matrices. More specifically these layers can be defined as (Durkan et al., 2019):

1. Given an input \mathbf{x} , split \mathbf{x} into two parts $\mathbf{x}_{1:d-1}$ and $\mathbf{x}_{d:D}$;
2. Using a neural network, compute parameters for a bijective function f using one half of \mathbf{x} : $\theta_{d:D} = \text{NN}(\mathbf{x}_{1:d-1})$; parameters $\theta_{1:d-1}$ are learnable parameters that do *not* depend on the input;
3. The output \mathbf{y} of the layer is then $y_i = f_{\theta_i}(x_i)$ for $i = 1, \dots, d_x$.

These coupling transforms thus act elementwise on their inputs.

Appendix I. Network Architectures and Hyperparameters

I.1 RQS flows and Bijecta

Within all Rational Quadratic Spline (RQS) flows we parameterise 4 knots for each spline transform. The hyper-parameters of the knots were as in the reference implementation from Durkan et al. (2019), available at github.com/bayesiains/nsf: we set the minimum parameterised gradient of each knot to be 0.001, the minimum bin width between each encoded knot and the origin to be 0.001, and the minimum height between each knot and the origin to be 0.001.

Unlike in Durkan et al. (2019), where they view a single RQS ‘layer’ as composed of a sequence of numerous coupling layers, in this paper the number of layers we describe a model as having is exactly the number of coupling layers present. So for our 4-layer models there are four rational quadratic splines. Each layer in our flows are composed of: an actnorm layer, an invertible 1x1 convolution, an RQS coupling transform and a final 1x1 invertible convolution.

The parameters of the knots were themselves parameterised using ResNets nets, as used in RealNVP Dinh et al. (2017), for each of which we used 3 residual blocks and batch normalisation. As in Dinh et al. (2017) we factor-out after each layer. All training was done using ADAM Kingma & Lei Ba (2015), with default β_1, β_2 , a learning rate of 0.0005 and a batch size of 512. We perform cosine decay on the learning rate during training, training for 25,000 steps.

Data was rescaled to 5-bit integers and we used RealNVP affine pre-processing so our input data was in the range $[\epsilon, 1 - \epsilon]$ with $\epsilon = 0.05$.

I.2 VAEs

Affine For the affine experiments we used fully convolutional encoders and decoders, each made out of 5 resnet block. Each block also down/up-scales their input by a factor of 2 along the spatial input dimensions. As our input images are 32×32 , this means that 5 such scalings map to 1×1 representations.

As we pass through the encoder we double the number of features for each block, from an initial number of 16. Thus the final residual output has $32 \times 16 = 512$ filters. A 1×1 convolutional layer then maps this to the posterior. The decoder performs these same operations in reverse order: an initial 1×1 convolutional layer maps the sampled value of the latent variable into a $1 \times 1 \times 512$ hidden representation, a chain of 5 upscaling resnet blocks map this to a $32 \times 32 \times 16$ representation. Finally a 1×1 convolution maps this to the $32 \times 32 \times 3$ sub-pixel means of the likelihood function.

CelebA For CelebA we used the same networks as used by Chen et al. (2018) for their CelebA experiments. The encoder and decoder are composed for 5 convolutions/transposed convolutions, with batchnorm layers in between. The number of filters increases as we go up the encoder: $64 \rightarrow 64 \rightarrow 128 \rightarrow 128 \rightarrow 512$ before being mapped to the posterior distribution’s parameters by a 1×1 convolution. The decoder has the same sequence of filter sizes in its transposed convolutions but in reverse, with a final convolution to the $32 \times 32 \times 3$ sub-pixel means of the likelihood function.

Appendix J. Reconstructions, Latent Traversals, and Samples

J.1 Numerical Results

Table 2: Here we evaluate the source-separation and reconstruction quality of non-square ICA models. We evaluate source separation by evaluating the mean log probability of the validation set embeddings in \mathcal{S} under our heavy-tailed prior, normalised by the dimensionality of \mathcal{S} space: $\log p(\mathbf{s})/d_s$ ($d_s = 64$). As our base distribution is heavy-tailed, this metric evaluates the axis-alignment, the *independence*, of learnt factors. We also give the validation set bits-per-dim (bpd), a scaled ELBO and thus a measure of model quality (lower better). Bijecta models consistently have lower bpd than the linear models. We evaluate the quality of low-dimensional representations by measuring the L_1 reconstruction error in \mathcal{X} . A better representation encodes more information in \mathcal{S} , making it easier for the model to then reconstruct in \mathcal{Z} and subsequently in \mathcal{X} . Most striking is the improvement across all metrics when introducing a *single* bijective mapping. Our 4-layer model further improves the quality of the compressed representations as seen by the lower reconstruction errors.

		CIFAR-10	MNIST	fashion-MNIST	CelebA
Linear-ICA	$\log p(\mathbf{s})/d_s$	-4.8	-4.29	-3.0	-9.0
	bits-per-dim	6.3	8.6	6.6	6.2
	L_1 reconstruction error in \mathcal{X}	3.0	3.1	2.9	2.9
1-layer Bijecta	$\log p(\mathbf{s})/d_s$	-4.2	-2.8	-2.2	-7.3
	bits-per-dim	3.8	3.2	3.8	3.4
	L_1 reconstruction error in \mathcal{X}	2.0	1.0	1.6	1.9
4-layer Bijecta	$\log p(\mathbf{s})/d_s$	-4.7	-3.0	-2.2	-7.8
	bits-per-dim	3.2	2.1	3.1	3.2
	L_1 reconstruction error in \mathcal{X}	1.9	0.6	1.2	1.4

J.2 Reconstructions



Figure J.3: Original data and reconstructions for Fashion MNIST (a)-(d), CIFAR-10 (e)-(h) and CelebA (i)-(l). The first column – (a), (e), (i) – shows a random selection of 40 images from each dataset’s training set. The second column – (b), (f), (j) – shows the reconstructions obtained by linear non-square ICA using our approximately-Stiefel unmixing matrix as is Appendix E, with $d_s = 64$. The third column – (c), (g), (k) – shows the reconstructions for Bijecta with an RQS flow with a single layer and $d_s = 64$. The fourth column – (d), (h), (l) – for a 4-layer Bijecta, $d_s = 64$. Bijecta models show much higher fidelity reconstructions than the linear ICA model. On CelebA the 4-layer model gives the highest fidelity reconstructions.

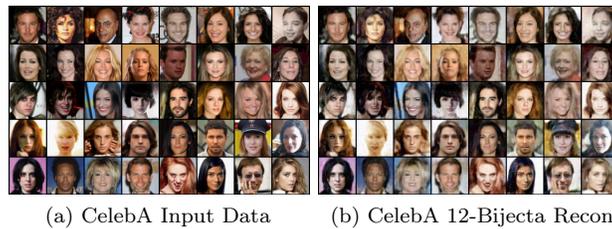
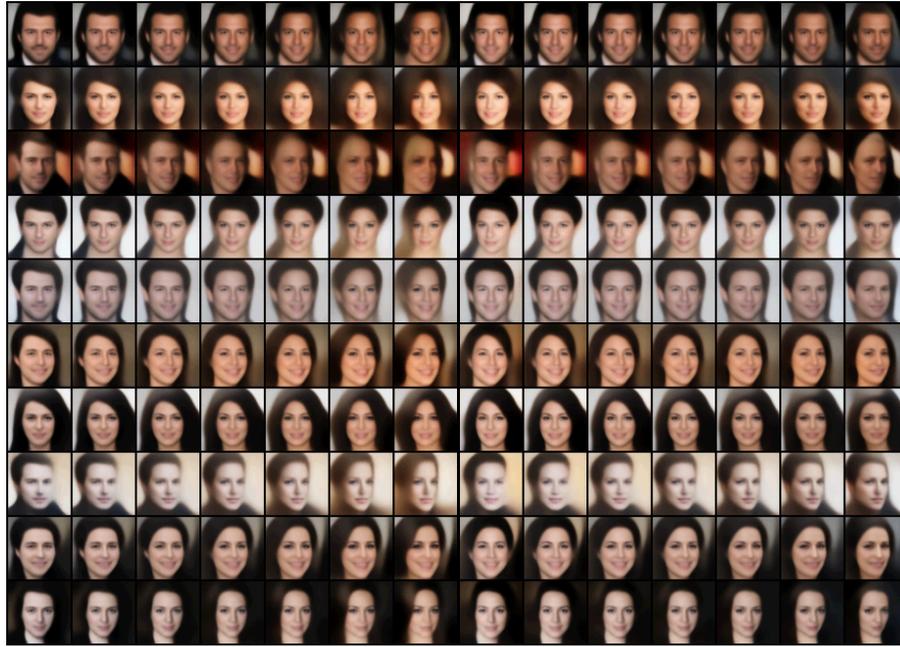


Figure J.4: Here we show reconstructions for a 12-layer Bijecta model, trained on a batch size of 64 for the CelebA dataset, with a latent space dimensionality $d_s = 512$. The quality of the reconstructions clearly illustrate that as we stack invertible layers our model and increase the size of \mathcal{S} we are able to reconstruct images with a high degree of accuracy.

J.3 Latent Traversals



(a) Dim 3 - Gender

(b) Dim 12 - Face Rotation



(c) Dim 15 - Light Warmth

(d) Dim 24 - Eye Shadow

(e) Dim 27 - Face Width

Figure J.5: Here we show latent traversals along 5 of 32 of the latent dimensions for 10 posterior samples from an 8-layer Bijecta model. We list the dimension's encoded factor of variation under each image.

J.4 Samples

J.4.1 MARGINAL POSTERIOR SAMPLES CELEBA



Figure J.6: Here we show decodings from an 8-layer Bijecta ($d_s = 32$) trained on CelebA where we sample from a factorised approximation to Bijecta's posterior.

J.4.2 PRIOR SAMPLES CELEBA

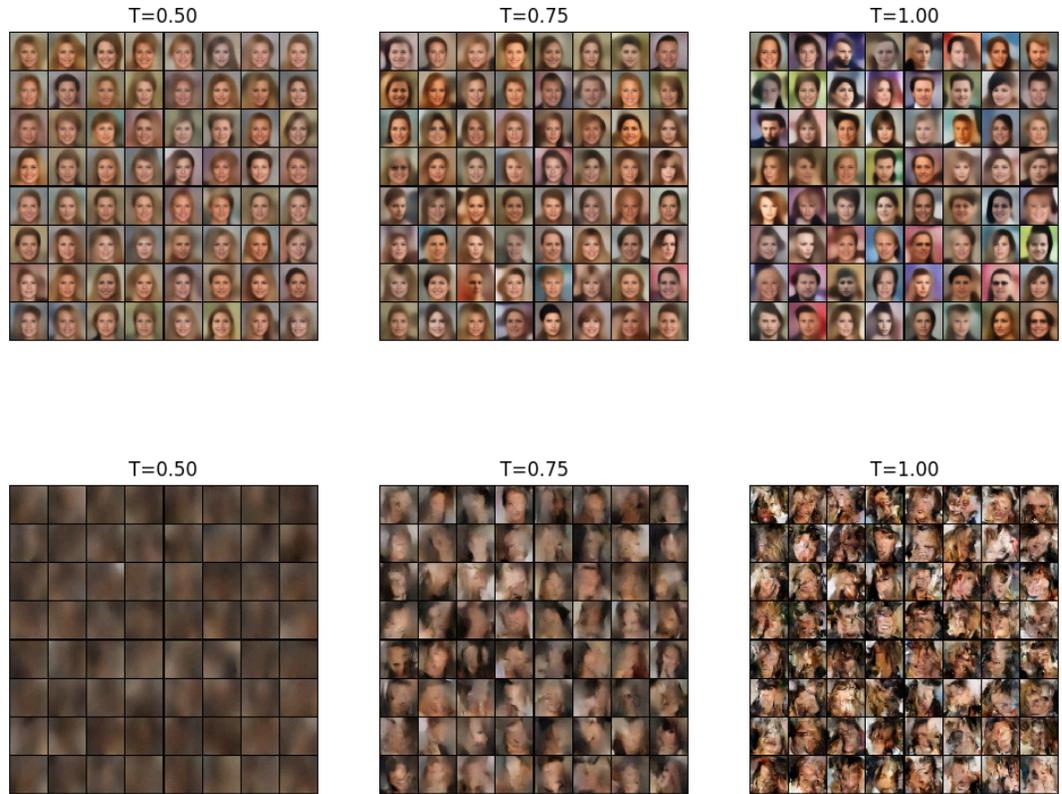


Figure J.7: Here we show samples from an 8-layer Bijeta model (top) and an 8-layer RQS-flow (bottom) with Laplace prior, trained on a batch size of 256 for the CelebA and CIFAR-10 datasets for 100000 steps, with a latent space dimensionality $d_s = 32$. We show samples rescaled at scale T.