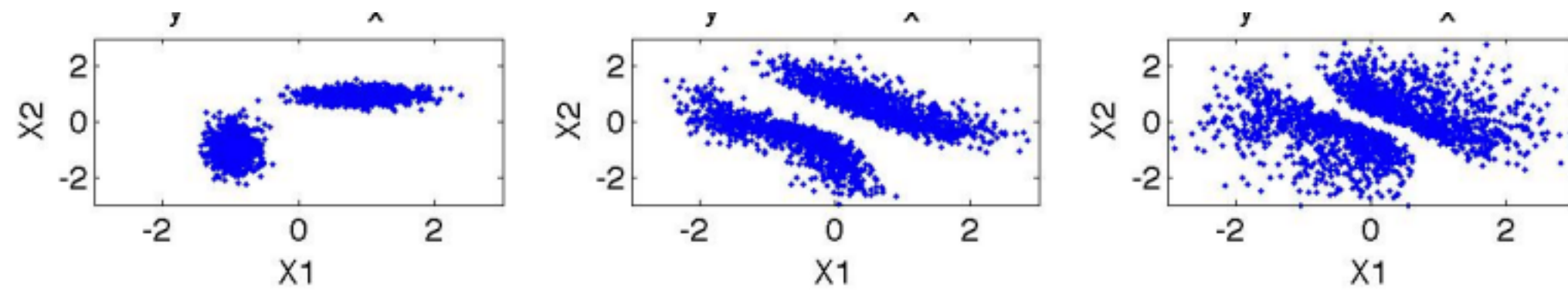# Normalizing Flow history

3.1 Regularizing the Transformations

We will train the model on data by minimizing an objective composed of several parts:

Divergence Penalty $\mathcal{D}(\Psi)$: This determines the fit of the current encoding transformation. It forces the marginal densities of the empirical distribution of the representation-space data to match a target distribution of our choice, by penalizing divergence from it.

Invertibility Measure $\mathcal{I}(\Theta)$: This ensures the invertibility of $f_\Theta(\cdot)$ by penalizing poorly-conditioned transformations.

Reconstruction Loss $\mathcal{R}(\Theta, \Psi)$: This jointly penalizes the encoder $g_\Psi(\cdot)$ and decoder $f_\Theta(\cdot)$ to ensure that $g_\Psi(y) \approx f_\Theta^{-1}(y)$ on the data.

Each of these participates in the overall objective given by:

$$C(\Theta, \Psi) = \mu_\mathcal{D}\mathcal{D}(\Theta) + \mu_\mathcal{I}\mathcal{I}(\Psi) + \mu_\mathcal{R}\mathcal{R}(\Theta, \Psi) \,, \qquad (6)$$

where $\mu_\mathcal{I}, \mu_\mathcal{D}, \mu_\mathcal{R} \in \mathbb{R}$ are the weights of each term. We will examine each of these terms in more detail in the proceeding sections.

2010
Tabak, Esteban G., and Cristina V. Turner.
"A family of nonparametric density estimation algorithms."

2013
Rippel, Oren, and Ryan Prescott Adams.
"High-dimensional probability estimation with deep density models."

General coupling layer Let $x \in \mathcal{X}$, I_1, I_2 a partition of $[\![1, D]\!]$ such that $d = |I_1|$ and m a function defined on $\mathbb{R}^d$, we can define $y = (y_{I_1}, y_{I_2})$ where:

$$y_{I_1} = x_{I_1}$$
$$y_{I_2} = g(x_{I_2}; m(x_{I_1}))$$

where $g : \mathbb{R}^{D-d} \times m(\mathbb{R}^d) \to \mathbb{R}^{D-d}$ is the *coupling law*, an invertible map with respect to its first argument given the second. The corresponding computational graph is shown Fig 2. If we consider $I_1 = [\![1, d]\!]$ and $I_2 = [\![d, D]\!]$, the Jacobian of this function is:

$$\frac{\partial y}{\partial x} = \begin{bmatrix} I_d & 0 \\ \frac{\partial y_{I_2}}{\partial x_{I_1}} & \frac{\partial y_{I_2}}{\partial x_{I_2}} \end{bmatrix}$$

Where I_d is the identity matrix of size d. That means that $\det \frac{\partial y}{\partial x} = \det \frac{\partial y_{I_2}}{\partial x_{I_2}}$. Also, we observe we can invert the mapping using:

$$x_{I_1} = y_{I_1}$$
$$x_{I_2} = g^{-1}(y_{I_2}; m(y_{I_1}))$$

We call such a transformation a *coupling layer* with *coupling function* m.

Additive coupling layer For simplicity, we choose as coupling law an *additive coupling law* $g(a; b) = a + b$ so that by taking $a = x_{I_2}$ and $b = m(x_{I_1})$:

$$y_{I_2} = x_{I_2} + m(x_{I_1})$$
$$x_{I_2} = y_{I_2} - m(y_{I_1})$$

2014
Dinh, Laurent, David Krueger, and Yoshua Bengio.
"Nice: Non-linear independent components estimation."