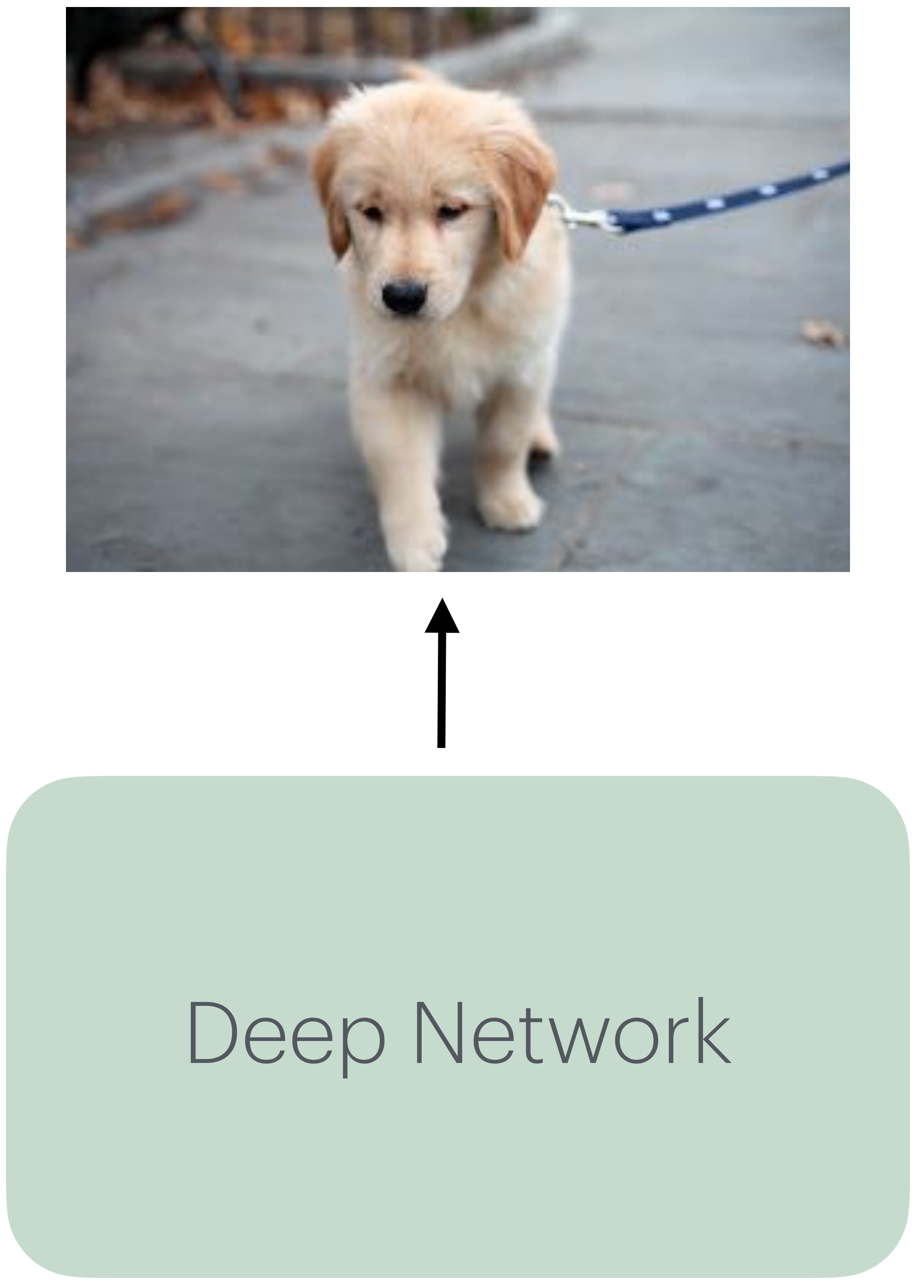
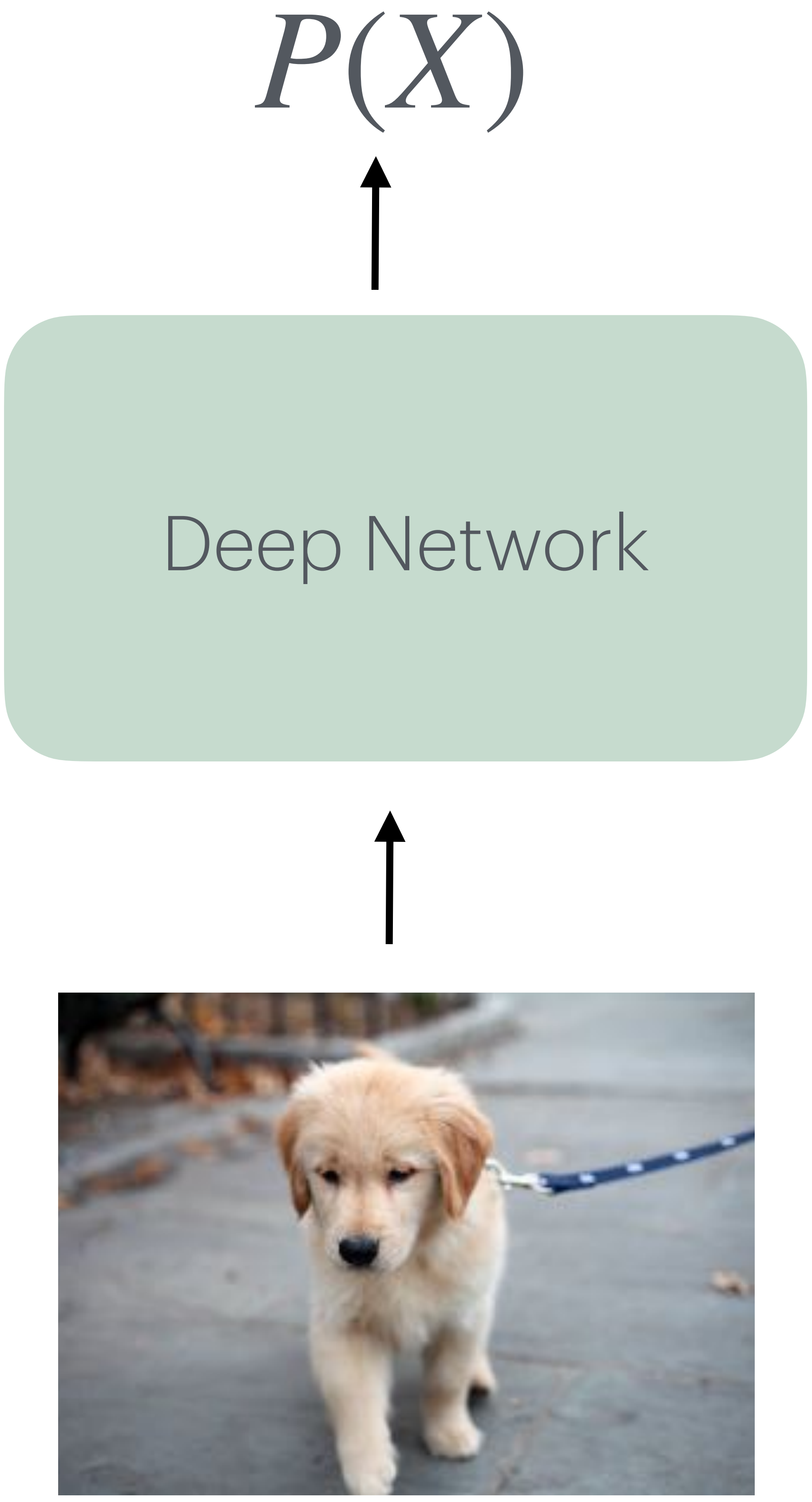
Flow-based models

Philipp Krähenbühl, UT Austin

Generative models

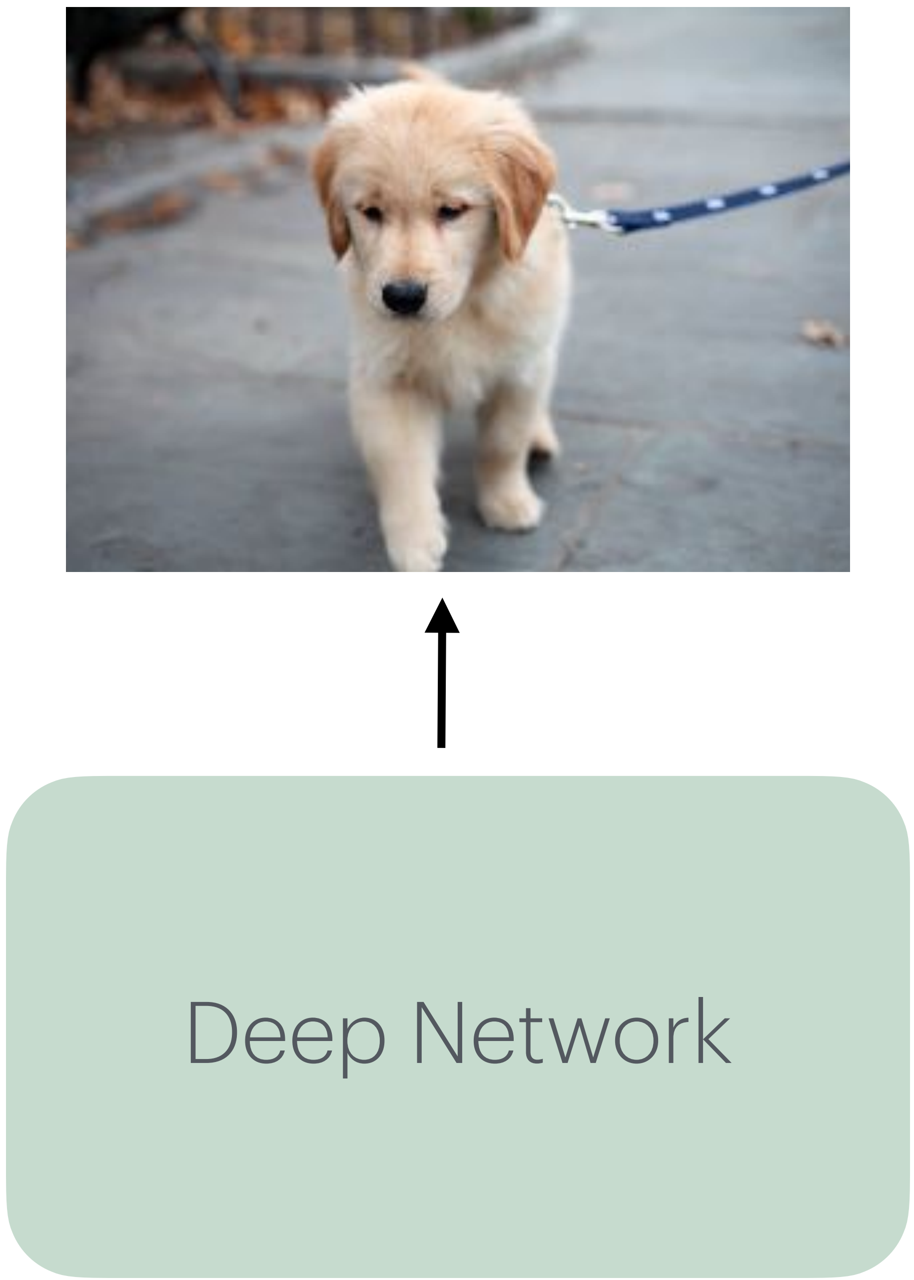
- Two tasks of a generative model $P(X)$
  - Sampling: $x \sim P(X)$
  - Density estimation: $P(X = x)$

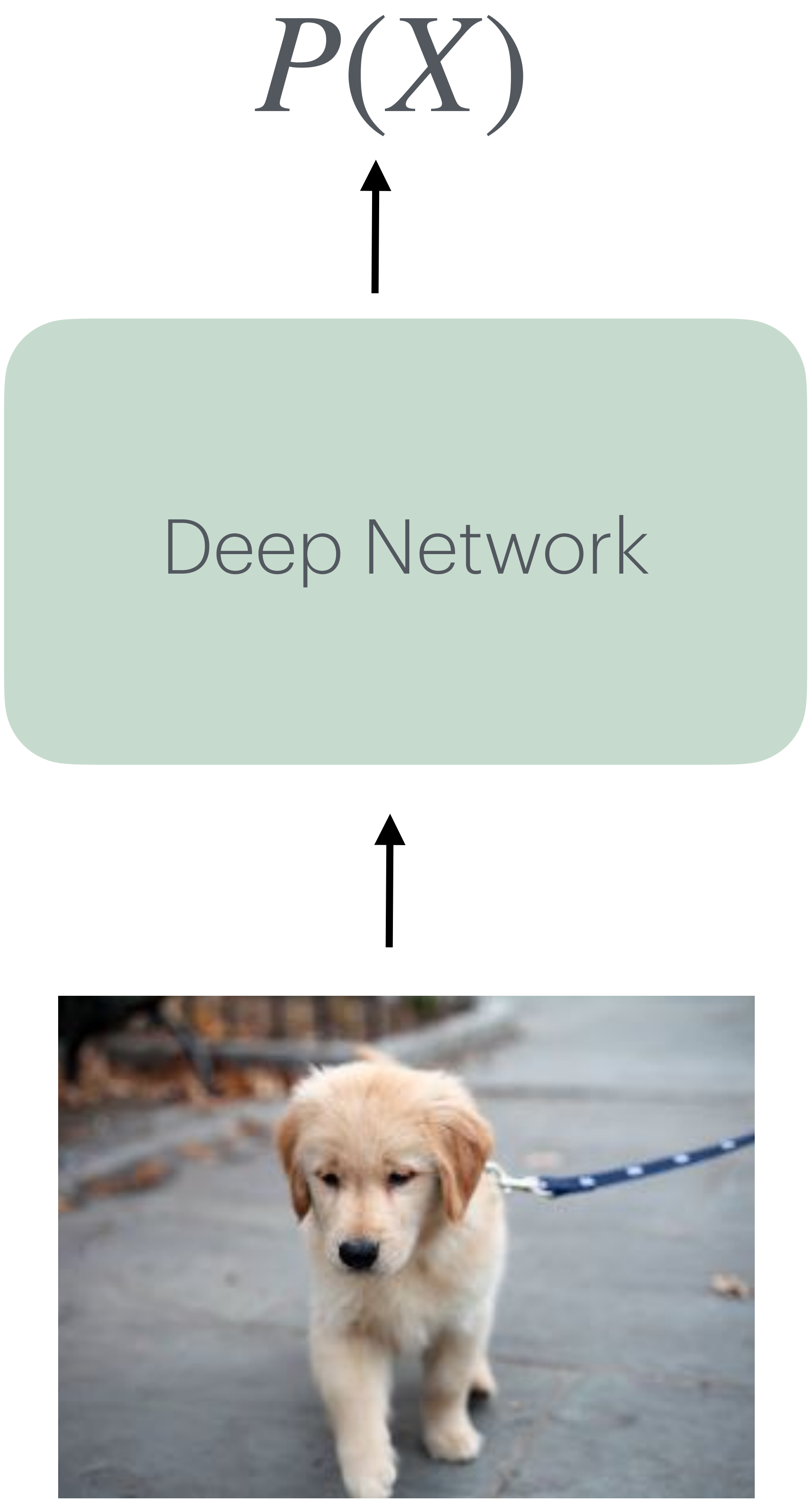




Generative modeling is hard

- Density estimation $P(X = x)$
  - How to ensure $\sum_x P(x) = 1$ for all x
  - Impossible to compute (in general)
- Sampling $x \sim P(X)$
  - What is the input to the network?



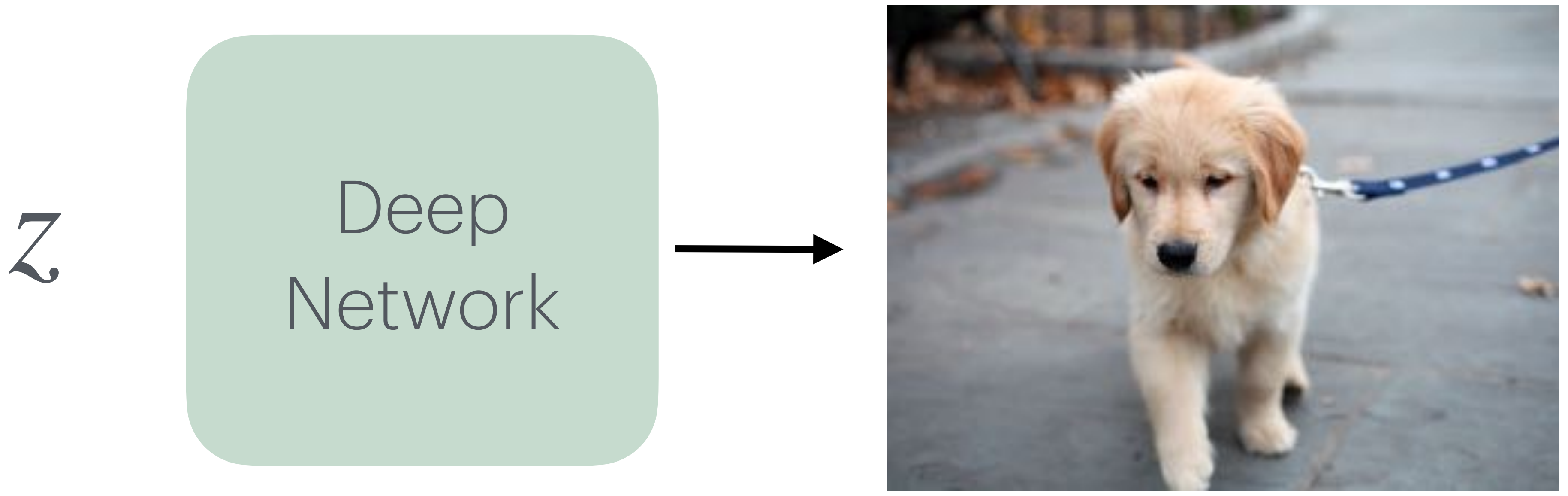


Generative models

Two kinds of models

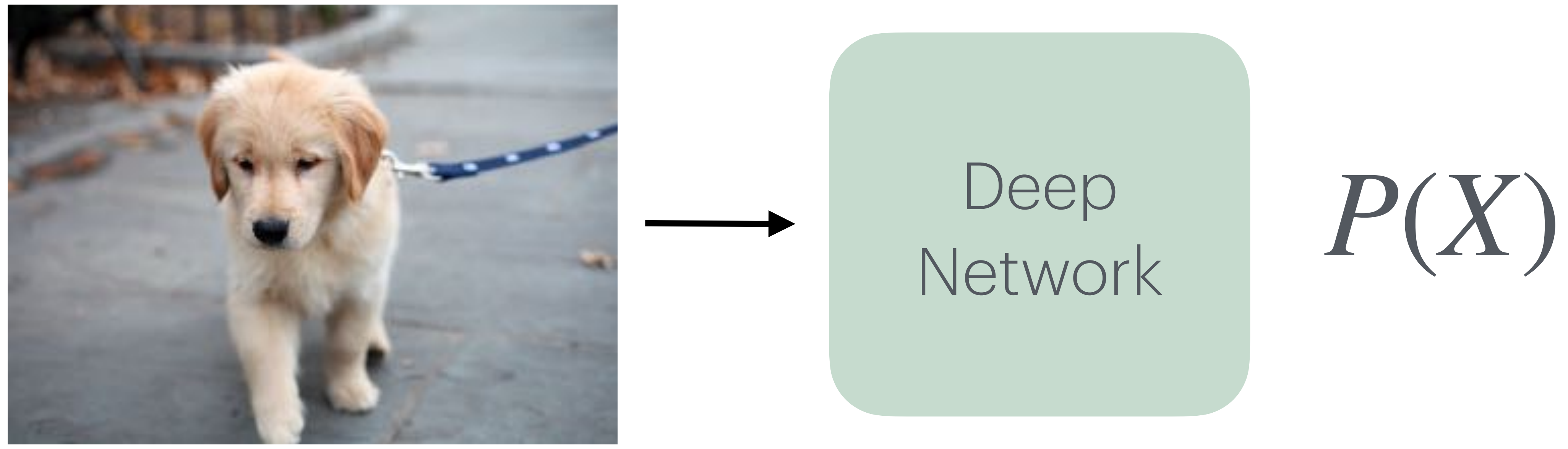
Sampling based $x \sim P(X)$

- Sample $z \sim P(Z)$
- Learn transformation
  - $P(x \mid z)$ or $f : z \rightarrow x$



Density estimation based $P(X)$

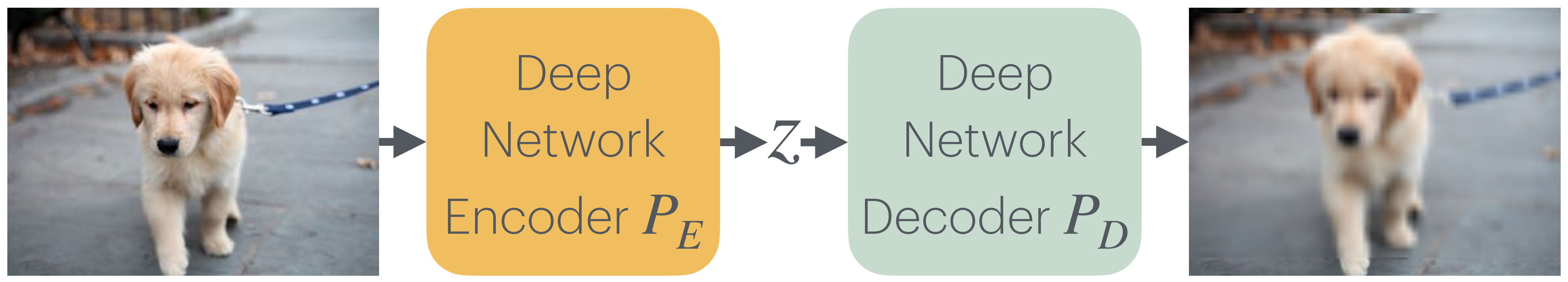
- Learn special form of $P(X)$
- Model specific sampling / generation



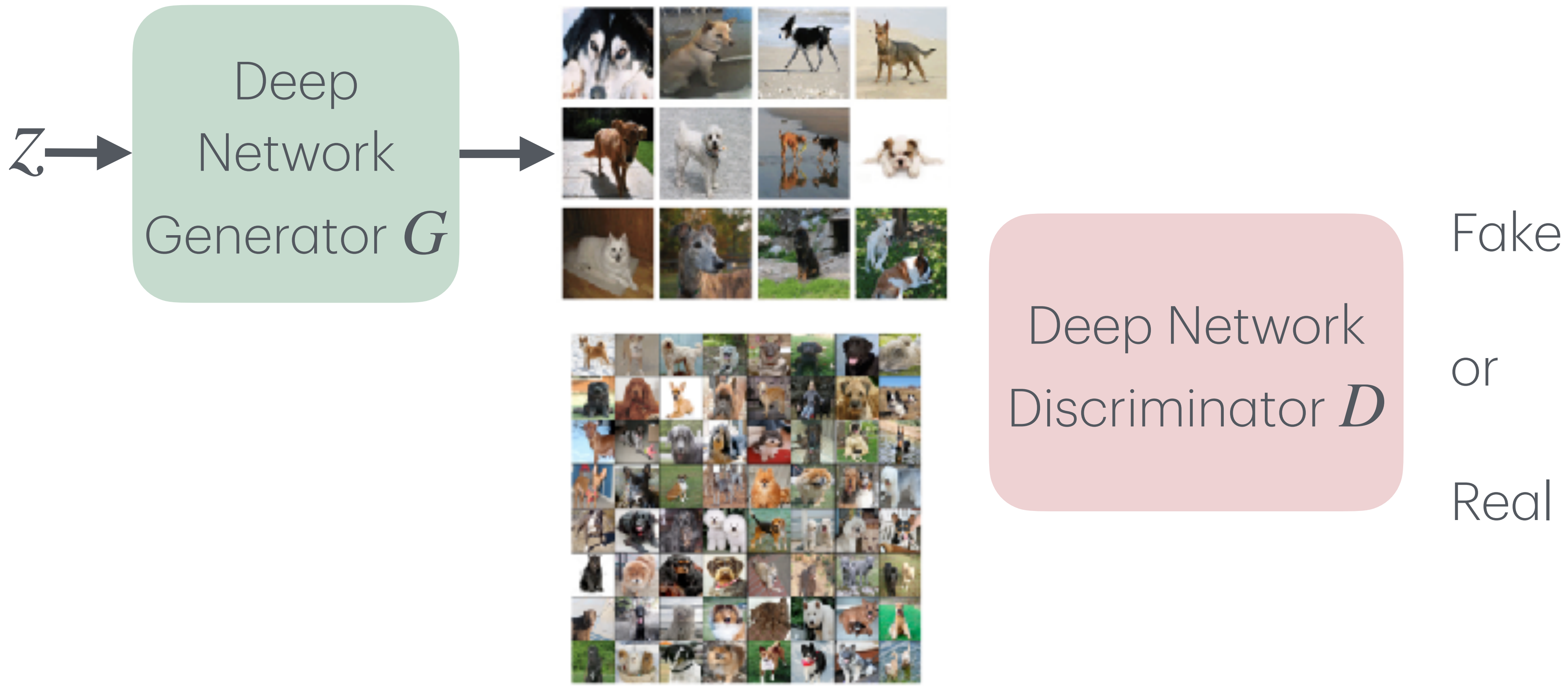
Recap

- VAE
  - Image -> latent space -> Image
  - Loss encourages Gaussian latent
- GAN
  - Gaussian -> Image
  - Loss compares distributions

Variational Auto Encoder (VAE)



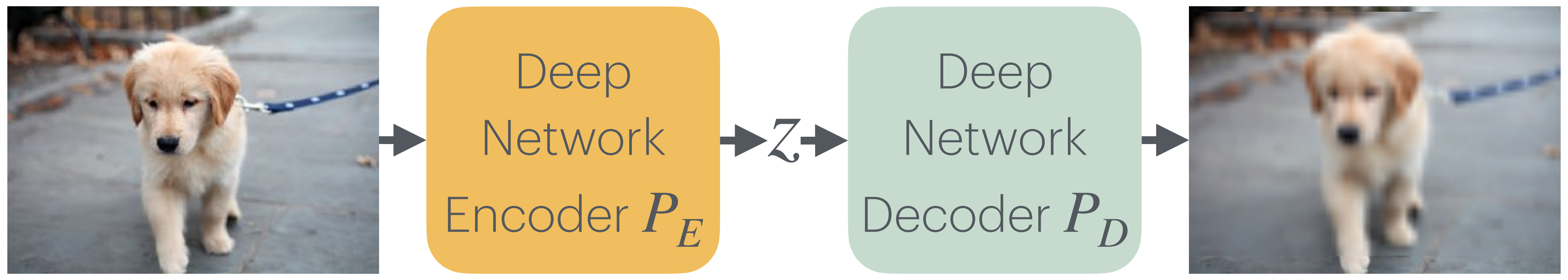
Generative Adversarial Network (GAN)



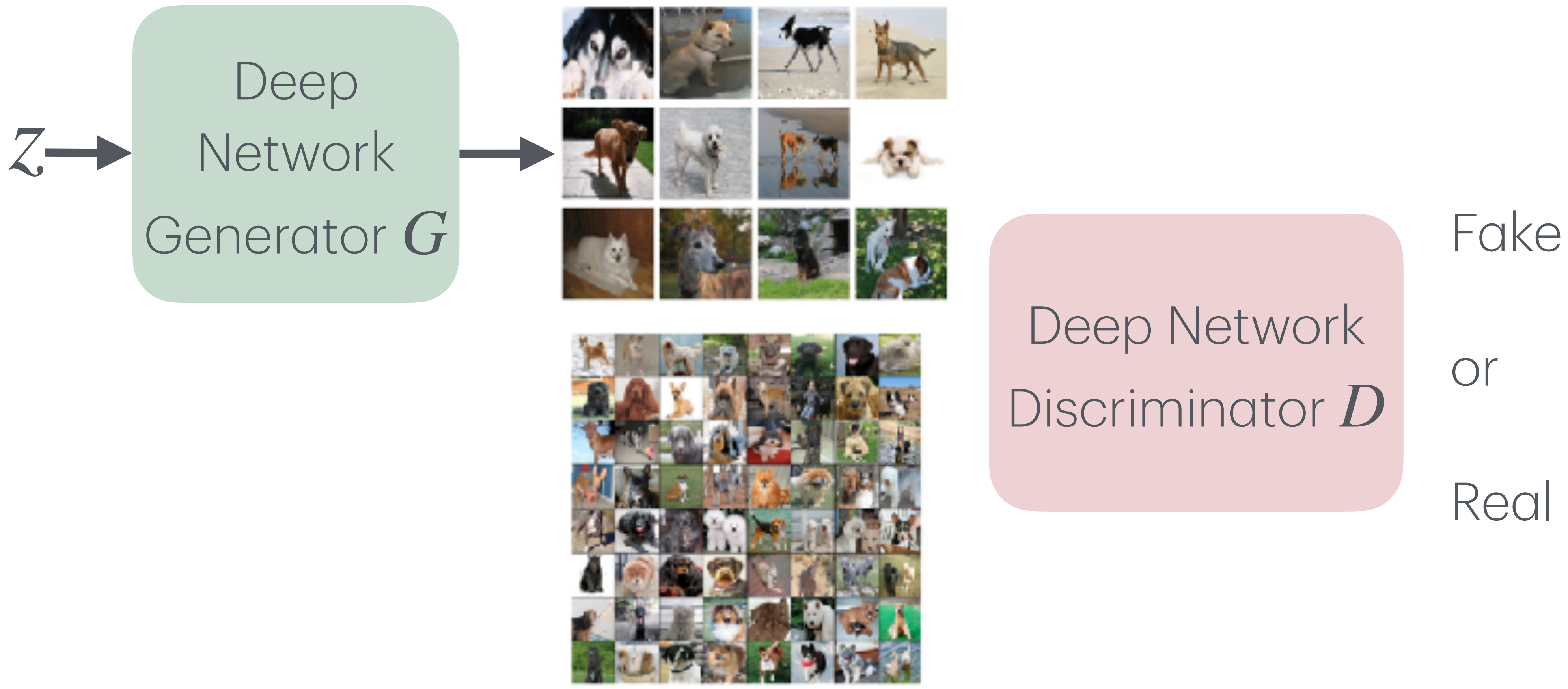
Flow-based models

- Assume generative $G : z \to x$ is invertible
  - G^{-1} exists and is easy to compute
  - By definition $G : \mathbb{R}^N \to \mathbb{R}^N$
- Define $P(Z) = \mathcal{N}(0,1)$
  - Compute $P(x)$
  - Maximize $\log P(x)$ on training images

Variational Auto Encoder (VAE)



Generative Adversarial Network (GAN)



Flow-based models

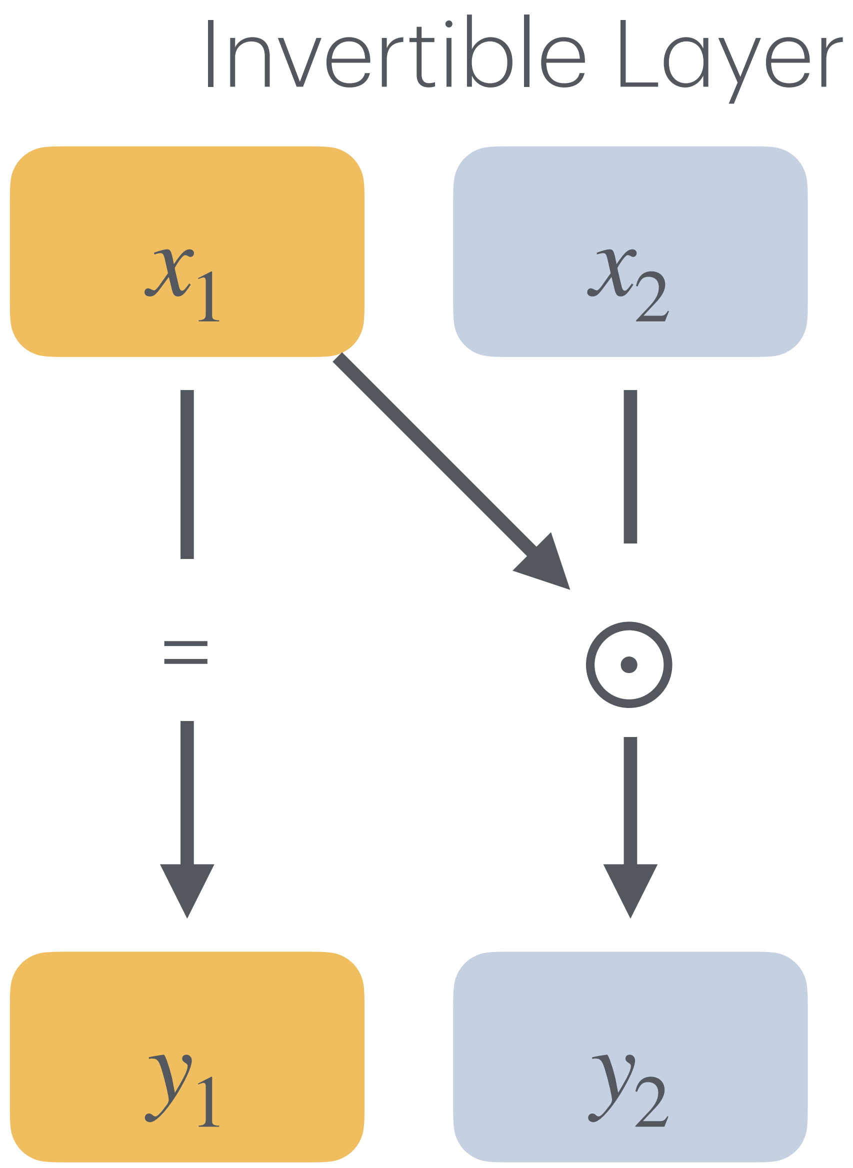
[1] Variational Inference with Normalizing Flows, Rezende etal 2015
[2] Density estimation using Real NVP, Dinh etal 2016
[3] Glow: Generative Flow with Invertible 1x1 Convolutions, Kingma etal 2018

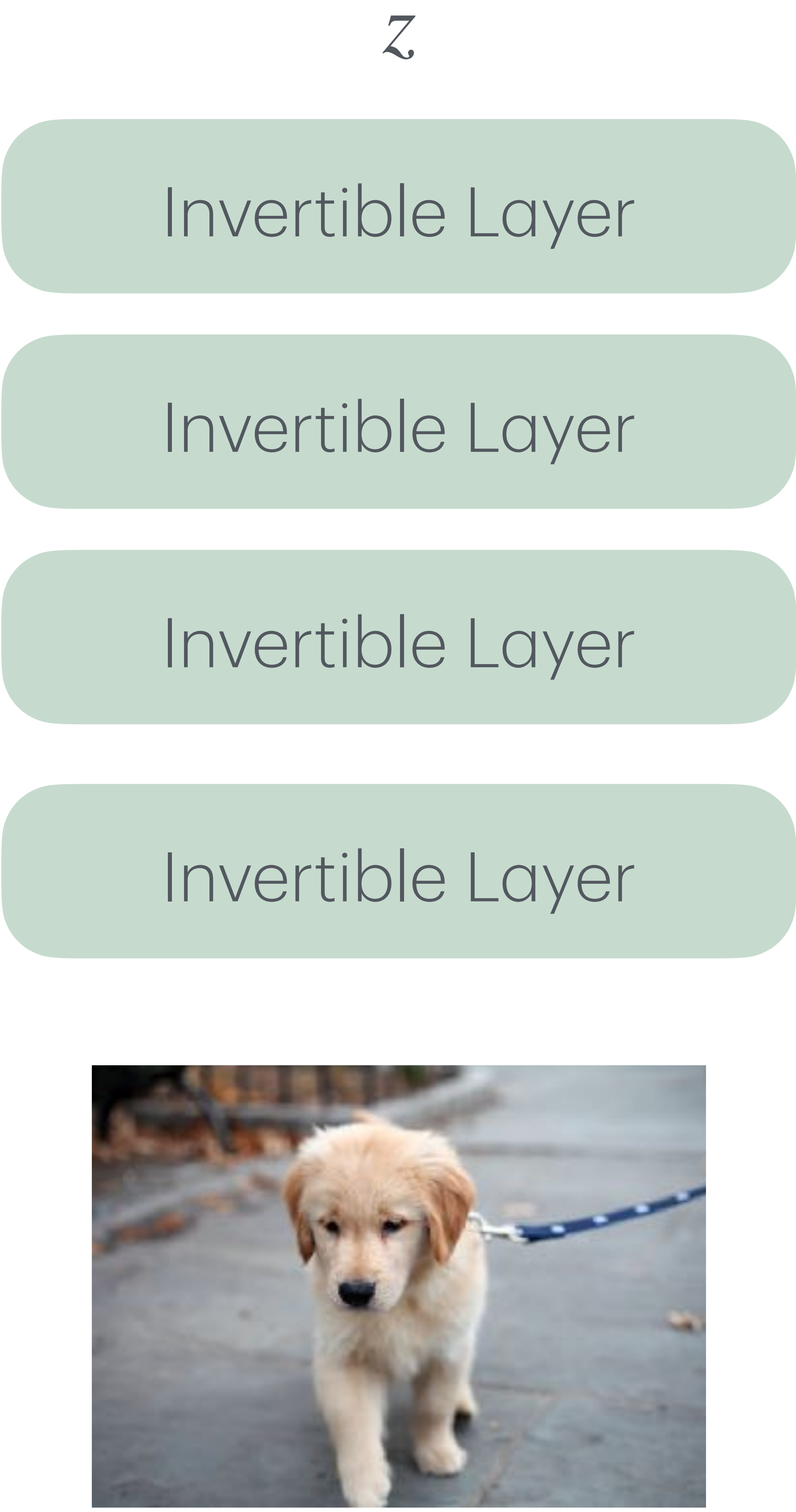
Flow-based models

Flow-based models

- Invertible $G : z \rightarrow x$ and $P(Z) = \mathcal{N}(0,1)$
- Change of variable formula
  - $P(x) = P(z) \left| det \left(\frac{\partial z}{\partial x} \right) \right| = P(G^{-1}(x)) \left| det \left(\frac{\partial G^{-1}(x)}{\partial x} \right) \right|$
  - Closed-form definition of $P(x)$
    - Only need invertible network and efficient determinant of Jacobean

Invertible Networks



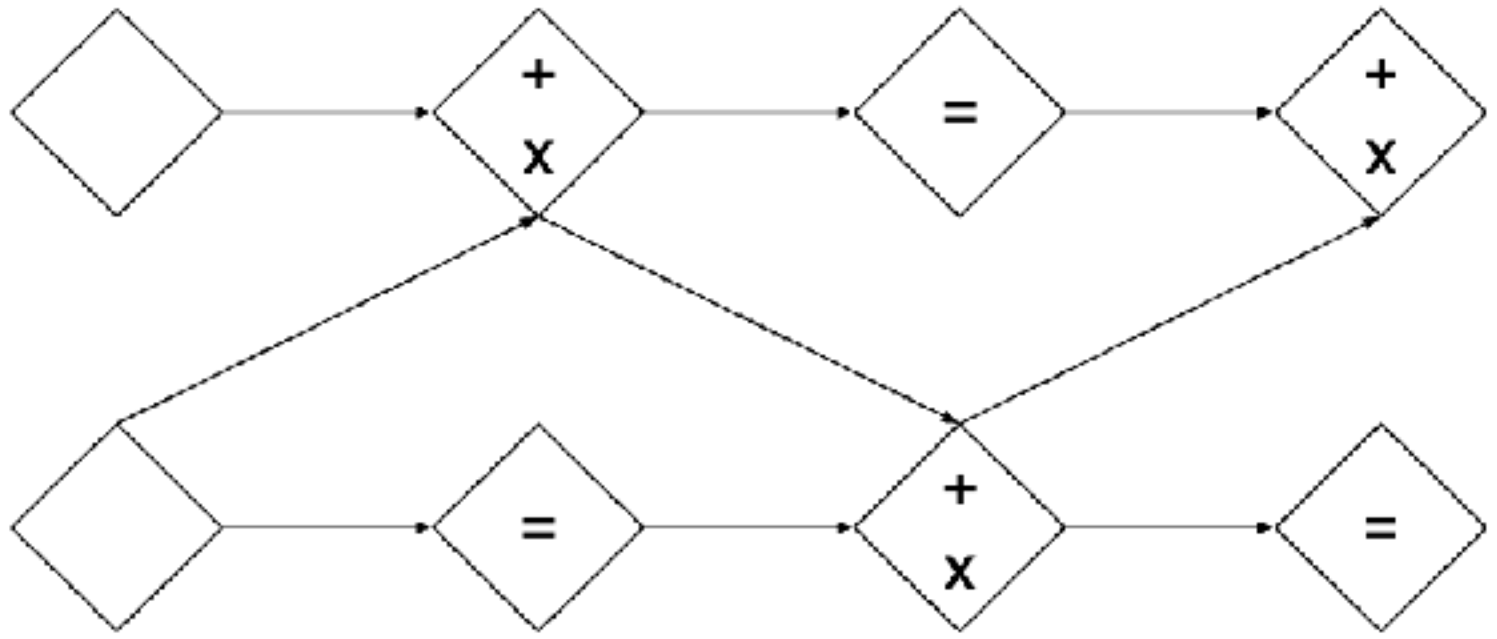


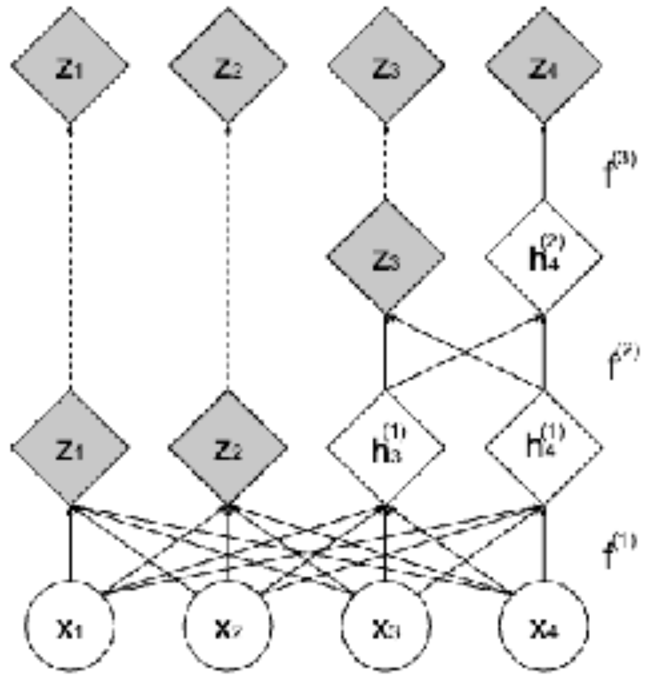
- Invertible Layer
  - Split inputs into two groups x_1, x_2
  - Split outputs into two groups y_1, y_2
  - $y_1 = x_1 \qquad y_2 = \exp(s(x_1)) \odot x_2 + t(x_1)$
  - Inverse

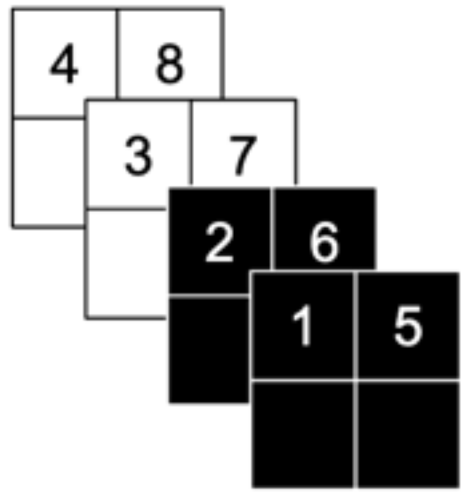
    $x_1 = y_1 \qquad x_2 = \exp(-s(y_1)) \odot (y_2 - t(y_1))$

[1] Variational Inference with Normalizing Flows, Rezende etal 2015
[2] Density estimation using Real NVP, Dinh etal 2016
[3] Glow: Generative Flow with Invertible 1x1 Convolutions, Kingma etal 2018

Invertible Networks







- Overall architecture
  - Alternate copy path
  - Coarse to fine (for efficiency)
  - Masked convolutions
  - Invertible 1x1 layers

[1] Variational Inference with Normalizing Flows, Rezende etal 2015
[2] Density estimation using Real NVP, Dinh etal 2016
[3] Glow: Generative Flow with Invertible 1x1 Convolutions, Kingma etal 2018

Flow-based models

- Generation: $z \sim N(0,1) \quad x = G(z)$
- Very good results
- Stable training
- Very restrictive architecture
  - Invertible layers

Flow-based models

[1] Variational Inference with Normalizing Flows, Rezende etal 2015
[2] Density estimation using Real NVP, Dinh etal 2016
[3] Glow: Generative Flow with Invertible 1x1 Convolutions, Kingma etal 2018

Generative models

Two kinds of models

Sampling based $x \sim P(X)$

- Sample $z \sim P(Z)$
- Learn transformation
  - $P(x \mid z)$ or $f : z \rightarrow x$

z Deep Network 

Density estimation based $P(X)$

- Learn special form of $P(X)$
- Model specific sampling / generation

 Deep Network $P(X)$

References

- [1] Variational Inference with Normalizing Flows, Rezende et al. 2015.
- [2] Density estimation using Real NVP, Dinh et al. 2016.
- [3] Glow: Generative Flow with Invertible 1x1 Convolutions, Kingma et al. 2018.