1. **INTRODUCTION**

- Neural networks usually model the conditional distribution $p(y|x)$, where $y$ denotes a label and $x$ features.
- Generative models, on the other hand, represent the distribution over features $p(x)$.
- Can we efficiently combine the two in a hybrid model of the joint distribution $p(y, x)$?

Invertible generative models (a.k.a. normalizing flows) are a broad class of models defined via the change-of-variables formula. An initial density $p(x)$ flows through a series of transformations $f(x)$ and morphs into some (usually simpler) prior distribution $p(z)$.

### Invertible Generative Models (Normalizing Flows)

#### Generalized Linear Models (GLMs)

Generalized linear models (GLMs) model the expected response (or label) $y$ as a transformation of the linear model $\beta^T z$ where $\beta$ are parameters and $z$ features (covariates).

$$\mathbb{E}[y|z] = g^{-1}(\beta^T z)$$

- **Regression**: $\mathbb{E}[y|z] = \text{identity}(\beta^T z)$
- **Binary Classification**: $\mathbb{E}[y|z] = \text{logistic}(\beta^T z)$

2. **BACKGROUND**

3. **COMBINING DEEP GENERATIVE MODELS AND LINEAR MODELS**

We define a model of the joint distribution $p(y, x)$ by instantiating a GLM on the output of a normalizing flow:

$$p(y|z; \theta) = p(y|z; \beta, \phi) p(z; \phi)$$

In practice, we add a weight to the flow terms to tradeoff between predictive and generative behavior:

$$\mathcal{L}(\theta) = \sum_{i=1}^{N} \left( \log p(y_i|x_i; \beta, \phi) + \lambda \log p(x_i; \phi) \right)$$

For a Gaussian prior on the GLM, the predictive model can be trained via the closed-form marginal likelihood:

$$\log p(y|f(x; \phi)) = \log N(y; 0, \sigma^2 + \lambda^{-1} z^T z)$$

4. **SIMULATION**

- **1D regression task with heteroscedastic noise**.
  - *Subfigure (a)* shows a Gaussian process and *Subfigure (b)* shows our Bayesian DIGLM. *Subfigure (c)* shows $p(x)$ learned by the same DIGLM (black line) and compares it to a KDE (gray shading).

5. **EXPERIMENTS**

- **Regression on Flight Delay Data Set** (N=5 million, D=8)
  - **Model**: DIGLM
  - **RMSE**: 6.91
  - **NLL**: 5.07
  - **This data set exhibits covariate shift between the train and test splits.**
  - **The DIGLM’s $p(x)$ component is able to detect this shift (see left).**

- **Classification on MNIST and SVHN**
  - **$\lambda$ controls the trade-off between $p(y|x)$ and $p(x)$.**
  - **Hybrid model is better able to detect the OOD inputs via $p(x)$.**

- **Semi-Supervised Learning: MNIST and Half Moons**
  - **SSL (VAT) with only 1000 labels (2% of labeled data) achieves <1% error on MNIST.**

6. **SUMMARY**

We defined a neural hybrid model that can efficiently compute both predictive $p(y|x)$ and generative $p(x)$ distributions, in a single feed-forward pass making it a building block for downstream applications of probabilistic deep learning.