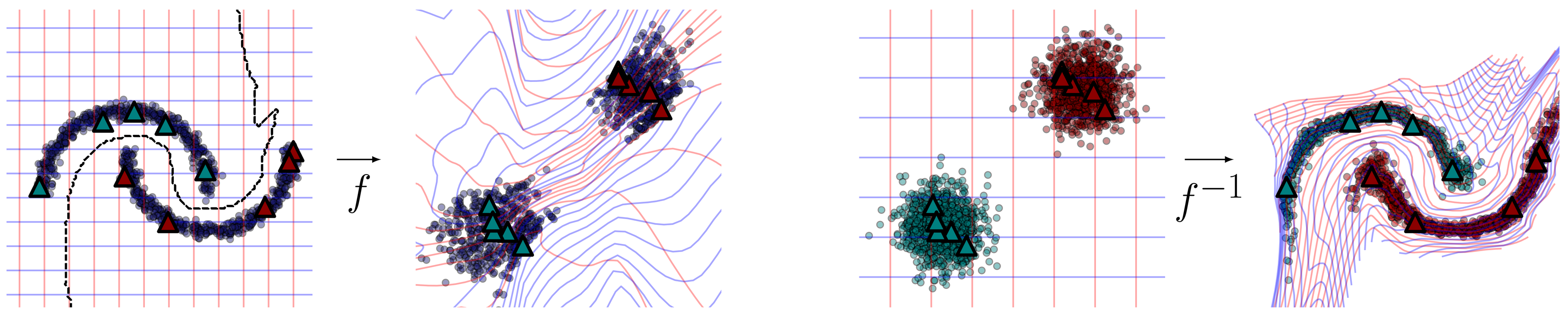
# Semi-Supervised Learning with Normalizing Flows

**Pavel Izmailov* Polina Kirichenko * Marc Finzi* Andrew Gordon Wilson**

**Cornell University**

We propose and study FlowGMM, a new classification model based on normalizing flows that can be naturally applied to semi-supervised learning. The idea of FlowGMM is to map each data class to a component in the Gaussian mixture using an invertible transformation. For semi-supervised learning:

- Labeled data from class $i$ is modeled as transformation of the $i$-th Gaussian
- Unlabeled data is modeled as transformation of the mixture



Observed Latent Latent Observed

**Figure 1:** Illustration of semi-supervised learning with Normalizing flows. Labeled data is shown with triangles, colored by the corresponding class label, and blue dots represent unlabeled data.

## FlowGMM

Define a normalizing flow with a class-conditional latent distribution

$$p_{\mathcal{X}}(x|y) = p_{\mathcal{Z}}\big(f(x)|y\big) \cdot \left|\frac{\partial f}{\partial x}\right|, \quad p_{\mathcal{Z}}(z|y) = \mathcal{N}(z|\mu_y, \Sigma_y).$$

We can evaluate likelihood for unlabeled data as

$$p_{\mathcal{X}}(x) = \frac{1}{\mathcal{C}}\sum_{k=1}^{\mathcal{C}} p_{\mathcal{X}}(x|y=k) = p_{\mathcal{Z}}\big(f(x)\big) \cdot \left|\frac{\partial f}{\partial x}\right|, \quad p_{\mathcal{Z}} = \frac{1}{\mathcal{C}}\sum_{k=1}^{\mathcal{C}} \mathcal{N}(\mu_k, \Sigma_k).$$

**Loss.** Log-likelihood for labeled $\mathcal{D}_l$ and unlabeled $\mathcal{D}_u$ data is

$$\log p_{\mathcal{X}}(\mathcal{D}_\ell, \mathcal{D}_u) = \sum_{(x_i,y_i)\in\mathcal{D}_\ell} \log p_{\mathcal{X}}(x_i|y_i) + \sum_{x_j\in\mathcal{D}_u} \log p_{\mathcal{X}}(x_j).$$

**Consistency Loss Term.** Encourages the model to map small perturbations of the same unlabeled inputs to the same components of the mixture:

$$L_{\text{cons}}(x', x'') = \mathcal{N}(f(x')|\mu_{y''}, \Sigma_{y''}),$$

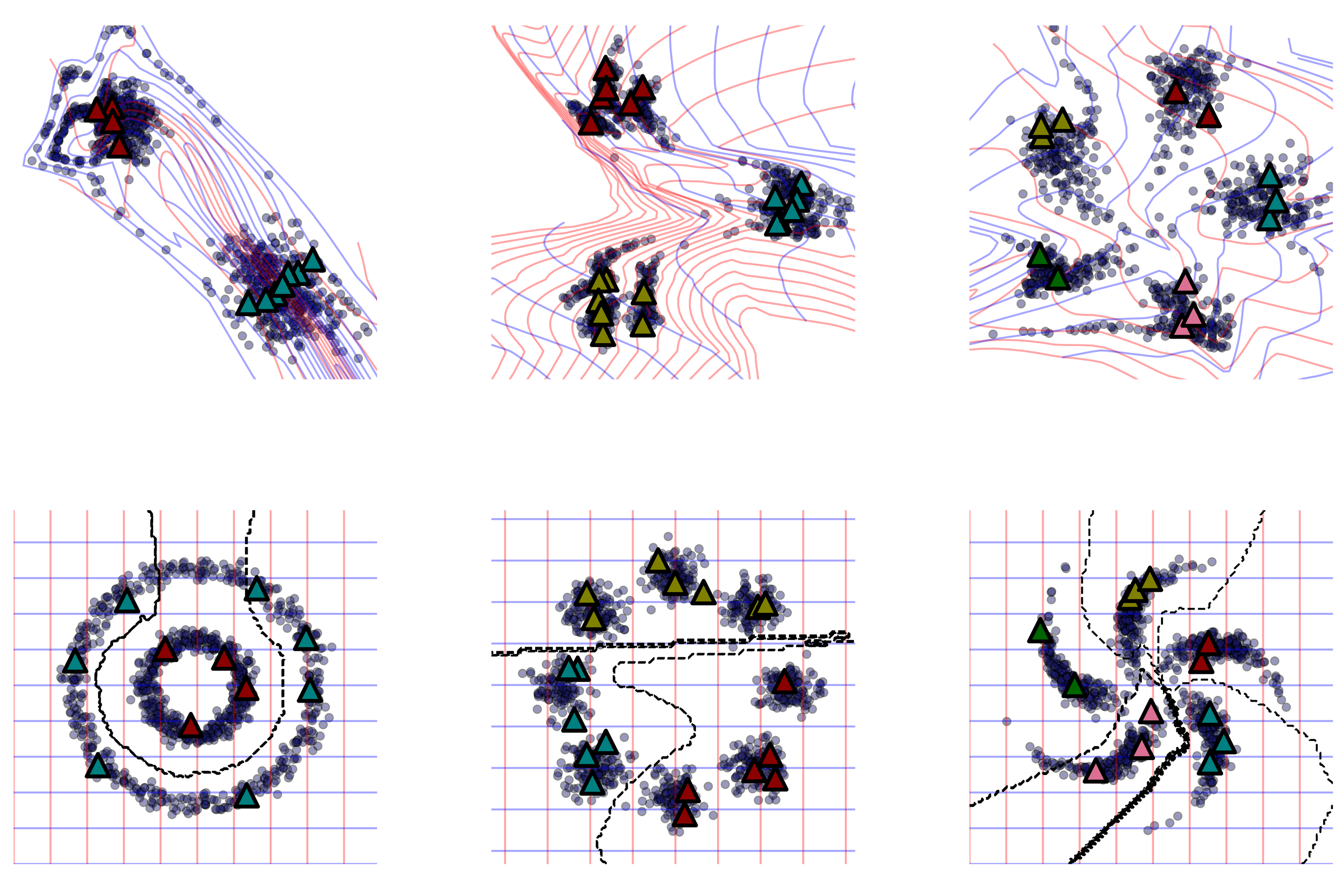
where $x'$ and $x''$ are two perturbations (e.g. random crops) of the same input $x$, and $y''$ is the class label predicted for $x''$.

**Classification.** Decision rule for a test point $x$:

$$y = \arg\max_{i\in\{1,\ldots,\mathcal{C}\}} p_{\mathcal{X}}(y=i|x) = \arg\max_{i\in\{1,\ldots,\mathcal{C}\}} \frac{\mathcal{N}(f(x)|\mu_i, \Sigma_i)}{\sum_{k=1}^{\mathcal{C}} \mathcal{N}(f(x)|\mu_k, \Sigma_k)}.$$

## Empirical Results

**Synthetic Data.** Even with a small number of labeled data points, FlowGMM puts the decision boundary to a low-density region in data-space.



(d) Two Circles (e) Eight Gaussians (f) Pinwheel

**Figure 2: Bottom**: unlabeled (blue dots) and labeled data (colored triangles) and decision boundary (dashed line). **Top**: mapping of the data to the latent space.

**Image Classifcation.** We use a Multiscale RealNVP architecture.

**Table 1:** Supervised and semi-supervised performance of the proposed model, VAE model (Kingma et al., 2014) and deep invertible generalized linear model (DIGLM, Nalisnick et al. 2019).

| Method | MNIST ($n_l = 1k, n_u = 59k$) | SVHN ($n_l = 1k, n_u = 72k$) | CIFAR-10 ($n_l = 4k, n_u = 46k$) |
|---|---|---|---|
| DIGLM Sup ($n_l + n_u$ labels) | 99.33 | 95.74 | - |
| FlowGMM Sup ($n_l + n_u$ labels) | 99.63 | 95.81 | 88.44 |
| M1+M2 VAE SSL ($n_l$ labels) | 97.60 | 63.98 | - |
| DIGLM SSL ($n_l$ labels) | 97.79 | - | - |
| FlowGMM Sup ($n_l$ labels) | 97.36 | 78.26 | 73.13 |
| FlowGMM ($n_l$ labels) | 98.94 | 82.42 | 78.24 |
| FlowGMM-cons ($n_l$ labels) | **99.0** | **86.44** | **80.9** |

**Uncertainty.** FlowGMM produces overconfident predictions on in-domain data; this problem can be remedied by scaling the variance of mixture components after the training is finished.

**Table 2:** Uncertainty calibration for FlowGMM trained on MNIST (1000 labeled objects) and CIFAR-10 in the supervised setting.

| | MNIST (test acc 97.3%) | | CIFAR-10 (test acc 89.3%) | |
|---|---|---|---|---|
| | FlowGMM | FlowGMM w Temp | FlowGMM | FlowGMM w Temp |
| NLL | 0.295 | 0.094 | 2.98 | 0.444 |
| ECE | 0.024 | 0.004 | 0.108 | 0.038 |

**Out-of-Domain Detection.** We use the likelihood $p_{\mathcal{X}}(x)$ of FlowGMM to identify out-of-domain data.

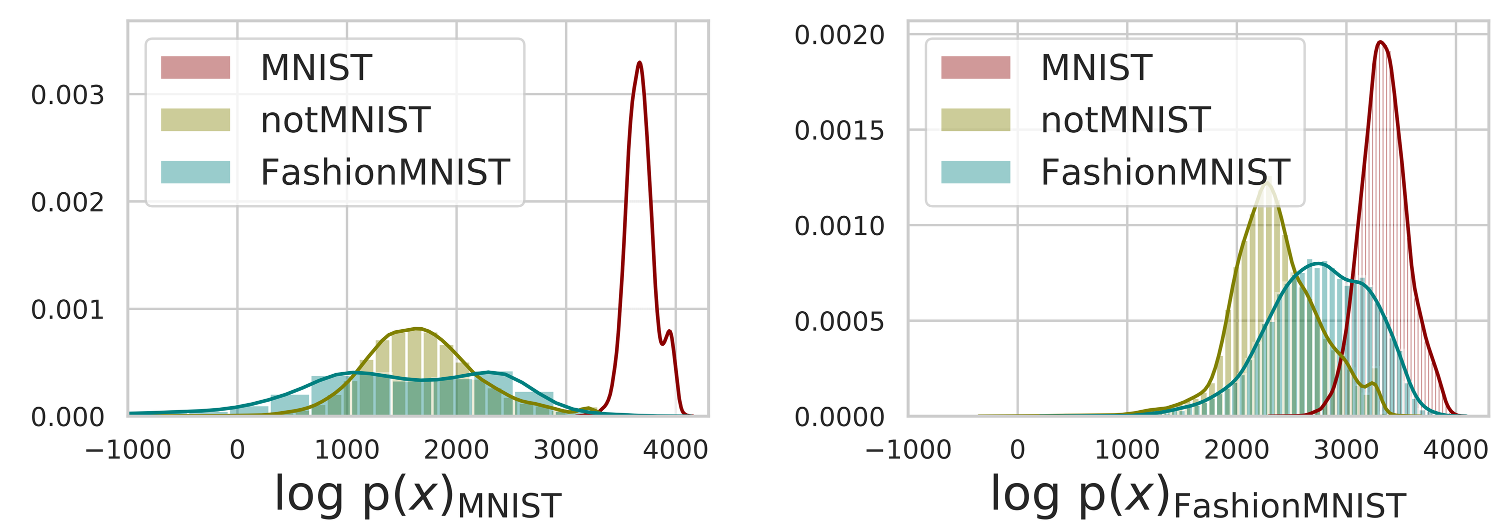



**Figure 3: Left:** Log-likelihoods on in- and out-of-domain data for our model trained on MNIST and **Right:** FashionMNIST.

- FlowGMM trained on MNIST can identify notMNIST and FashionMNIST data as out-of-domain
- On the other hand, MNIST examples are assigned higher likelihoods by our model trained on FashionMNIST than the training data itself

**Latent Representation.** FlowGMM naturally encodes the *clustering principle*: the decision boundary between classes must lie in the low-density region.

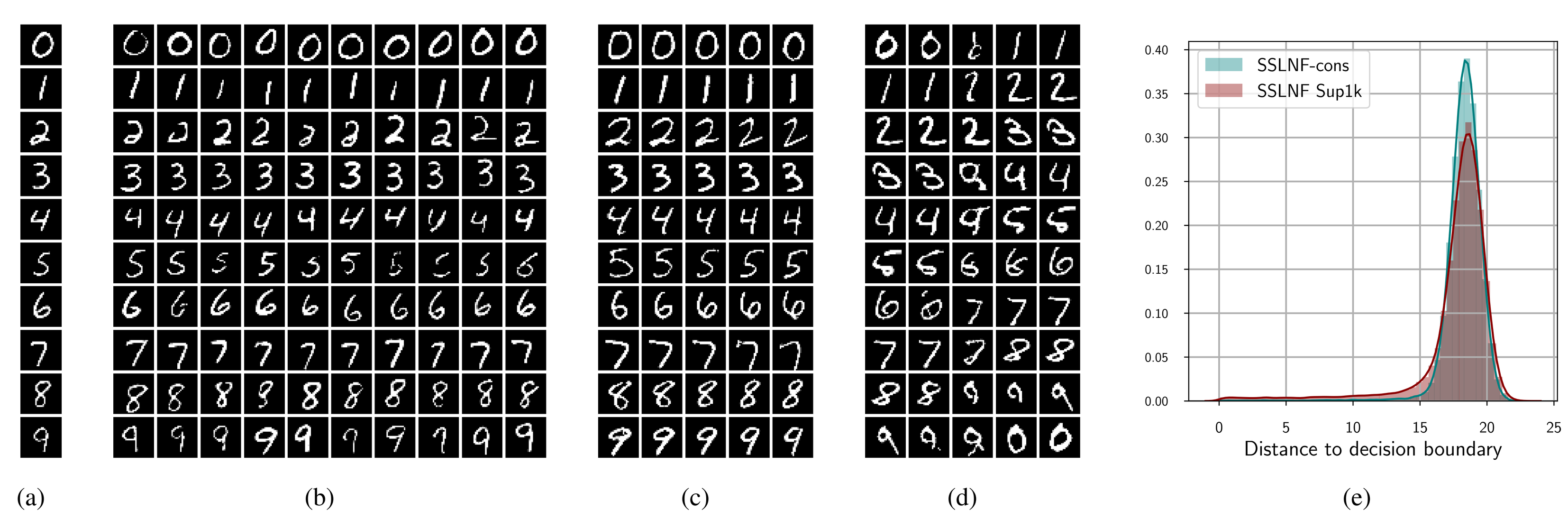



**Figure 4:** **(a)**: Images corresponding to means of the Gaussians for each class. **(b)**: Class-conditional samples from the model at a reduced temperature $T = 0.25$. **(c)**: Latent space interpolations between test images from the same class and **(d)**: from different classes. **(e)**: Histogram of distances from unlabeled data to the decision boundary for FlowGMM-cons trained on $1k$ labeled and $59k$ unlabeled data and FlowGMM Sup trained on $1k$ labeled data only.

- FlowGMM learns a reasonable generative model
- Interpolations between data points from different classes pass through low-density regions
- FlowGMM pushes the decision boundary away from unlabeled data