One-Line-of-Code Data Mollification Improves Optimization of Likelihood-based Generative Models

Ba-Hien Tran, Giulio Franzese, Pietro Michiardi, Maurizio Filippone

Objective and Contributions
- We verify that the success of score-based diffusion models (DMs) is in part due to the process of data smoothing, by incorporating this in the training of likelihood-based generative models (GMs), e.g., VAEs and normalizing flows
- Connecting this to continuation methods in the optimization literature
- Easy to implement by adding one line of code in any training loop!
- Showing consistent improvements in terms of quality of samples

Training Likelihood-based Generative Models
- Given a dataset \( D \overset{\Delta}{=} \{ x_i \}_{i=1}^N \), we aim to estimate the unknown data generating distribution \( p_{\text{data}}(x) \) by training a generative model \( p(x) \)
- Common approach to estimate \( \theta \) is to maximize the likelihood of the data

\[
L(\theta) \overset{\Delta}{=} - \mathbb{E}_{p_{\text{data}}(x)} [\log p(x)]
\]

Data Mollification
- Main idea: Adding Gaussian noise to the data throughout training and gradually reducing its variance until recovering the original data

\[
q(x_t | x) = \mathcal{N}(\tilde{x}_t; \alpha_t x, \sigma_t^2 I),
\]

where \( \alpha_t = \sqrt{1 - \gamma^2} \) and \( \sigma_t^2 = \gamma(t/T) \), with \( T \) is the maximum training iteration, and \( \gamma(.) \) monotonically decreases from 1 to 0 controlling the rate of smoothing

One Line of Code in Training Loop

```python
1 for t ← 1, 2, ..., T do
  2 x ~ p_{\text{data}}(x) // Sample training data
  3 \tilde{x}_t = \alpha_t x + \sigma_t \varepsilon // Smooth data with \( \alpha_t, \sigma_t^2 \overset{\Delta}{=} \gamma(t/T) \) and \( \varepsilon \sim \mathcal{N}(0, I) \)
  4 \theta_t ← \text{UPDATE}(\theta_{t-1}, \tilde{x}_t) // Train the model
```

Manifold Hypothesis and Manifold Overfitting
- Manifold hypothesis
  - Real-world high-dimensional data tend to lie on a manifold \( \mathcal{M} \) characterized by a much lower dimensionality
  - Data points on the manifold should be associated with high probability density, while points outside the manifold lie in regions of nearly zero density
- Manifold overfitting
  - The model \( p_\theta(x) \) assigns an arbitrarily large likelihood in the vicinity of the manifold, even if it does not capture accurately the data distribution \( p_{\text{data}}(x) \)
  - This makes it difficult for GMs to capture the true data distribution

Experiments on Image Datasets

<table>
<thead>
<tr>
<th>Dataset</th>
<th>( 20\times20 )</th>
<th>( 40\times40 )</th>
<th>( 64\times64 )</th>
<th>( 128\times128 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>CIFAR10</td>
<td>Vanilla</td>
<td>GAUSS. MOLLIFICATION</td>
<td>BLURRING</td>
<td></td>
</tr>
<tr>
<td>real-nvp</td>
<td>-13.88 ± 0.18</td>
<td>-13.88 ± 0.18</td>
<td>-13.88 ± 0.18</td>
<td>-13.88 ± 0.18</td>
</tr>
<tr>
<td>glow</td>
<td>-14.87 ± 0.36</td>
<td>-14.87 ± 0.36</td>
<td>-14.87 ± 0.36</td>
<td>-14.87 ± 0.36</td>
</tr>
<tr>
<td>real-nvp</td>
<td>-15.24 ± 0.38</td>
<td>-15.24 ± 0.38</td>
<td>-15.24 ± 0.38</td>
<td>-15.24 ± 0.38</td>
</tr>
<tr>
<td>glow</td>
<td>-15.69 ± 0.40</td>
<td>-15.69 ± 0.40</td>
<td>-15.69 ± 0.40</td>
<td>-15.69 ± 0.40</td>
</tr>
</tbody>
</table>

Densities Estimation on UCI Datasets

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Vanilla</th>
<th>GAUSS. MOLLIFICATION</th>
<th>BLURRING</th>
</tr>
</thead>
<tbody>
<tr>
<td>MNIST</td>
<td>101.30</td>
<td>101.30 ± 0.24</td>
<td>98.04 ± 0.12</td>
</tr>
<tr>
<td>BANES</td>
<td>71.38</td>
<td>71.38 ± 0.16</td>
<td>69.56 ± 0.14</td>
</tr>
<tr>
<td>UCI</td>
<td>71.38</td>
<td>71.38 ± 0.16</td>
<td>69.56 ± 0.14</td>
</tr>
</tbody>
</table>

References