Introduction

Variational Continual Learning

\[ q(\theta) = \arg \min_{q} \text{KL} \left( q(\theta) \mid \mid \frac{1}{Z} q_{t-1}(\theta) p(D_t | \theta) \right) \]

Continual Learning

- Continual Learning (CL) requires balance between:
  - Plasticity (Catastrophic Forgetting)
  - Stability (Inability to adapt)

Methods

Discriminative Model

Generative Model

- Architecture:
  - Fully-Connected
  - CNNs
- Coreset Memory Enhancement:
  - Random Coresets
  - K-Center Coresets
  - Variational Generative Replay (VGR)

Toy Task

Train T1/Test T1

Train T2/Test T1

Train T2/Test T2

Split-MNIST

Task 1

Task 2

Task 3

Task 4

Task 5

Task 6

Task 7

Task 8

Task 9

Task 10

- VCL avoids catastrophic forgetting
- VGR improves on explicit coresets
- CNN: complexity vs. performance

Permuted-MNIST

Task 1

Task 2

Task 3

Task 4

Task 5

Task 6

Task 7

Task 8

Task 9

Task 10

- VGR underperforms from unstructured images
- Performance plateaus with coreset size

References

1. Cuong V. Nguyen et al., Variational Continual Learning, 2018
2. Sebastian Farquhar et al., A Unifying Bayesian View of Continual Learning, 2019
3. Diederik P. Kingma et al., Variational Dropout and the Local Reparameterization Trick, 2015
4. Gido M. van de Ven et al., Generative replay with feedback connections as a general strategy for continual learning, 2019

Conclusion

- Variational methods improve over naive CL approaches
- VCL applies to both discriminative and generative models
- Coresets improve both naive and VCL performance
- Selection algorithm inconsequential
- VGR reduces coresets memory footprint
- VCL can be applied to CNNs
- CNN matches FC performance with fewer parameters
- LRT suboptimal for CNNs

Code available at: https://github.com/goldfarbDave/vcl