Semi-Supervised Learning: Core Methods

Proxy Label Methods: Overview

- Proxy label methods generate pseudo-labels for unlabeled data using model predictions.
- These pseudo-labels are treated as ground truth to improve the model iteratively.
- Two major approaches:
  - Self-Training
  - Co-Training

Self-Training (Detailed)

- Train a base classifier f on labeled data $\mathcal{D}_L$
- Predict labels on unlabeled data $\mathcal{D}_U$
- Select high-confidence predictions (e.g., $p > 0.95$) as pseudo-labels
- Add pseudo-labeled samples to $\mathcal{D}_L$, retrain

Example:

If a model predicts "digit 3" with 98

Loss

$$\mathcal{L} = \text{CrossEntropy}(f(x), y)$$

Co-Training (Detailed)

- Uses two classifiers f_1 and f_2 trained on two distinct views (e.g., left and right halves of an image)
- Each model labels examples for the other, assuming each view is sufficient and independent.
- Iterative training with pseudo-label sharing improves generalization.

Example:

In MNIST, one model trains on the left half, another on the right. Each pseudo-labels samples for the other.

Loss

$$\mathcal{L} = \mathrm{CE}(f_1(x^{(1)}), y) + \mathrm{CE}(f_2(x^{(2)}), y)$$

Variational Autoencoders (VAEs): Foundation

- A VAE is a generative model with a latent variable z and reconstruction objective.
- Learns $q(z|x)$ and generates $p(x|z)$, with a KL-divergence regularization.

VAE Loss (ELBO)

$$\mathcal{L}_{\text{VAE}} = \mathbb{E}_{q(z|x)}[\log p(x|z)] - D_{\text{KL}}(q(z|x) \| p(z))$$

VAE for SSL: Architecture

- Extends the VAE to condition on labels y (i.e., $q(z|x, y), p(x|z, y)$)
- A classifier $q(y|x)$ is added for handling unlabeled data
- Unlabeled loss is marginalized over all y:

Unlabeled Loss

$$\mathcal{L}_{\text{unsup}} = \sum_{y} q(y|x)[\mathbb{E}_{q(z|x,y)}[\log p(x|z, y)] - D_{\text{KL}}(q(z|x, y)\|p(z))]$$

Graph-Based SSL: Motivation

- Each data point is a node, edges reflect similarity
- Goal: Propagate labels from labeled to unlabeled nodes using graph structure
- Works best when data lies on a manifold or has cluster structure

Graph Construction Propagation

- Similarity can be defined via:
  - kNN (connect to k nearest neighbors)
  - RBF kernel: $w_{ij} = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right)$
- Label propagation iteratively updates:

$$F^{(t+1)} = \alpha S F^{(t)} + (1 - \alpha) Y$$

Graph Laplacian Regularization

Loss with Graph Smoothness

$$\mathcal{L} = \mathcal{L}_{\text{sup}} + \lambda \sum_{i,j} w_{ij} \|f(x_i) - f(x_j)\|^2$$

- Enforces label smoothness across similar data points
- Common in manifold regularization and GCNs

Conclusion

- Proxy methods bootstrap training using confident predictions
- VAEs enable joint generative-discriminative training in SSL
- Graph-based SSL is powerful for structured data and manifolds