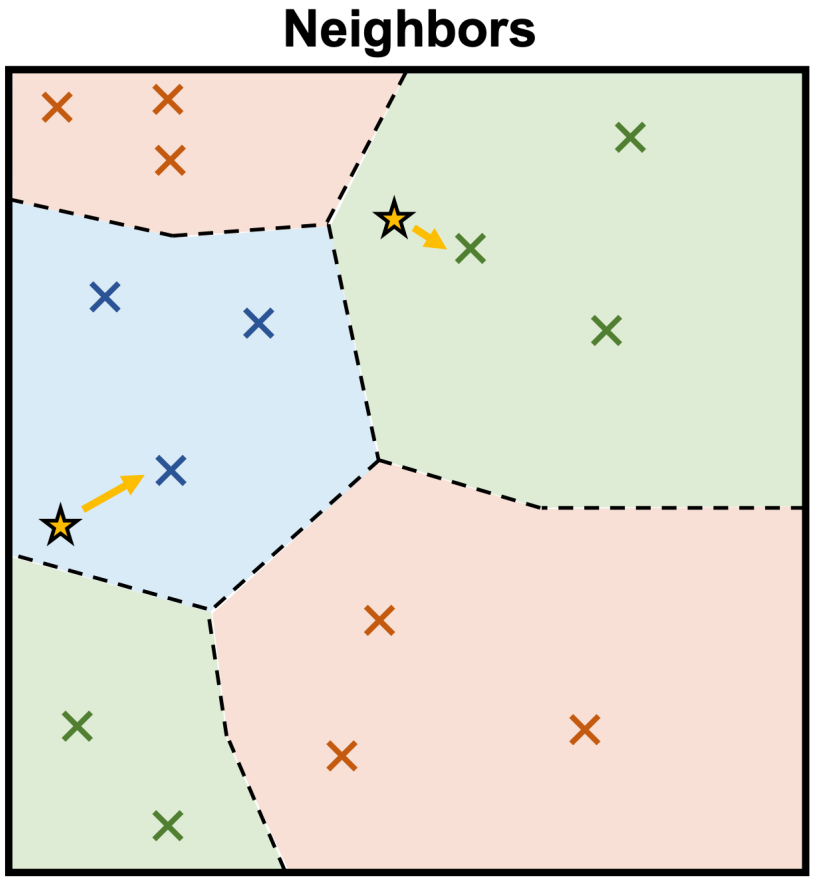
Infinite Mixture Prototypes for Few-Shot Learning

Kelsey Allen (MIT), Evan Shelhamer (Berkeley),

Hanul Shin (MIT), Josh Tenenbaum (MIT)

ICML 2019

Review: Neighborhood component analysis



OVERFITTING

Probability of a data point toward a neighbor

$$p_{ij} = \frac{\exp(\|Ax_i - Ax_j\|^2)}{\sum_{k \neq j} \exp(\|Ax_i - Ax_k\|^2)}. \tag{1}$$

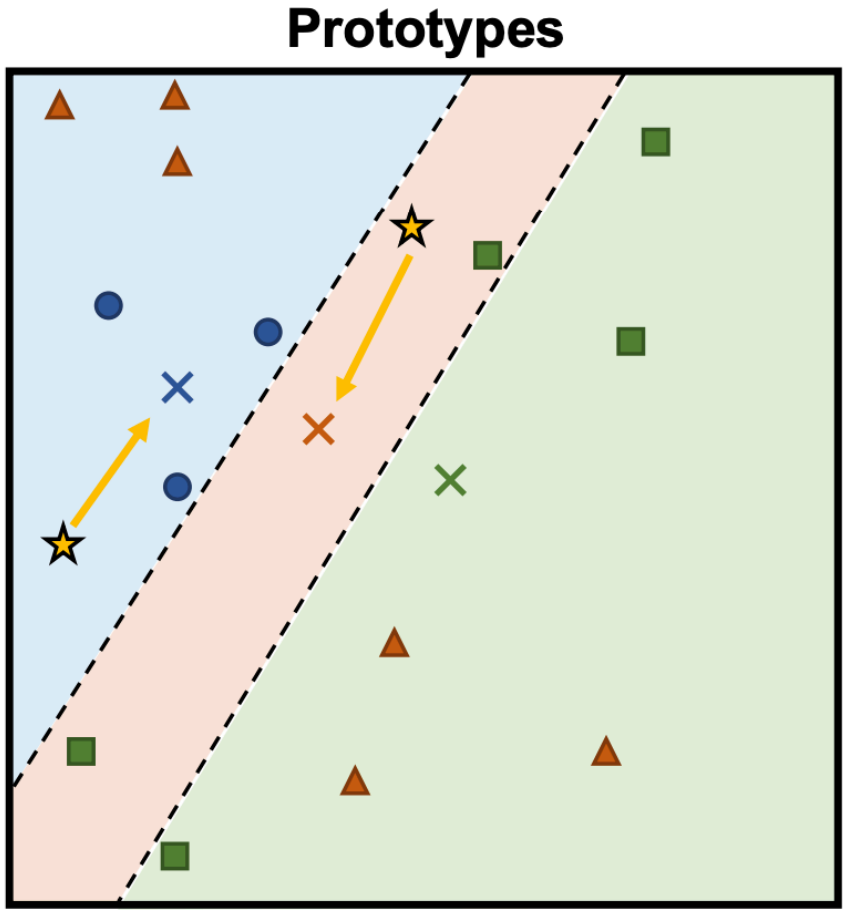
Classification as a marginal probability

$$p_A(y = n \,|\, x_i) = \sum_{j:y_j = n} p_{ij}. \tag{2}$$

Review: Prototypical network

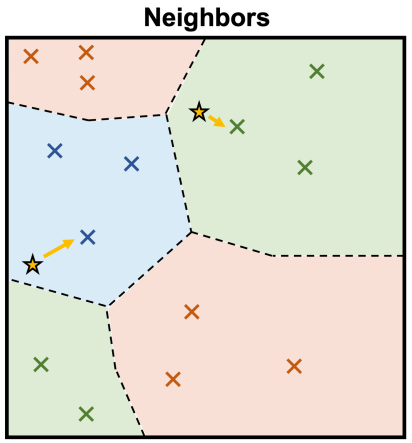
$$\mu_n = \frac{1}{|S_n|} \sum_{(x_i, y_i) \in S_n} h_\phi(x_i), \tag{3}$$

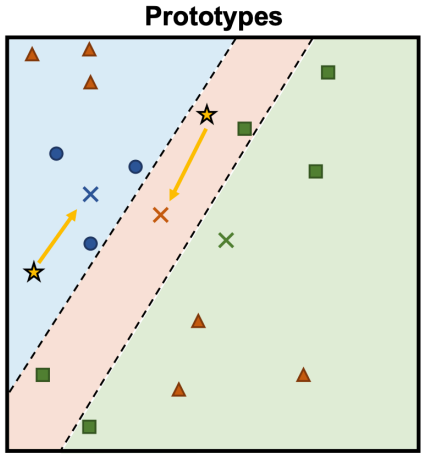
$$p_\phi(y' = n \,|\, x') = \frac{\exp(-d(h_\phi(x'),\, \mu_{c_n^*}))}{\sum_{n'} \exp(-d(h_\phi(x'),\, \mu_{c_{n'}^*}))} \tag{6}$$



UNDERFITTING

Two extremes





- Prototype/class: k
- Number of instances = Number of training samples

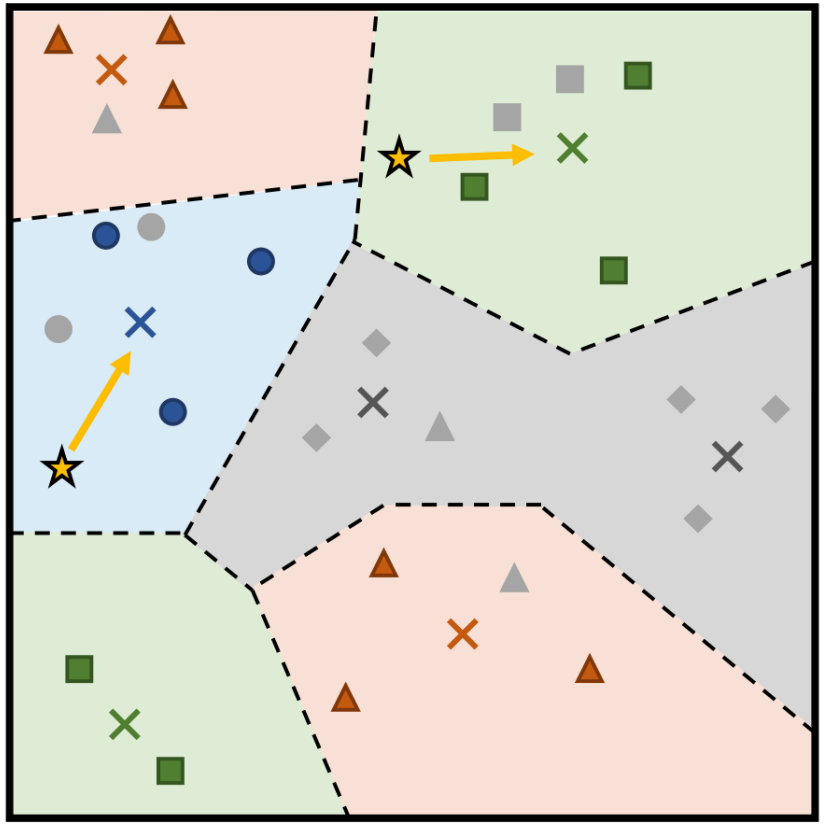
- Prototype/class: 1
- Number of instance = Number of class

Increase the number of prototypes/class

Reduce the number of instances

Review: Infinite Mixture Model

1. Init clusters based on training data
2. Calculate smallest distance from all points to all centroids
3. Make a new cluster if the distance is greater than threshold



Adaptive clustering algorithm based on distance threshold

$$\lambda = 2\sigma \log\left(\frac{\alpha}{(1+\frac{\rho}{\sigma})^{d/2}}\right) \qquad (5)$$

Where:

α: is the concentration param of Chinese Restaurant Process

ρ: measure of the standard deviation of the base distribution from which clusters are assumed to be drawn in the CRP

σ: cluster variance

Algorithm

Algorithm 1 IMP: support prototypes and query inference

Require: supports $(x_1, y_1)..., (x_K, y_K)$ and queries $x'_1, ..., x'_{K'}$
Return: clusters (μ_c, l_c, σ_c) and query classifications $p(y'|x')$

1. Init. each cluster μ_c with label l_c and $\sigma_c = \sigma_l$ as class-wise means of the supports, and C as the number of classes
2. Estimate λ as in Equation 5
3. Infer the number of clusters

for each point x_i **do**
  for c in $\{1, ..., C\}$ **do**

$$d_{i,c} = \begin{cases} \|h_\phi(x_i) - \mu_c\|^2 & \text{if } (x_i \text{ is labeled and } l_c = y_i) \\ & \text{or } x_i \text{ is unlabeled} \\ +\infty & \text{otherwise} \end{cases}$$

  end for
  If $\min_c d_{ic} > \lambda$: set $C = C + 1$, $\mu_C = h_\phi(x_i)$, $l_C = y_i$, $\sigma_C = \{\sigma_l$ if x_i labeled, σ_u otherwise$\}$.
end for

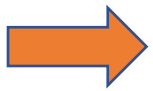
4. Assign supports to clusters by $z_{i,c} = \frac{\mathcal{N}(h_\phi(x_i); \mu_c, \sigma_c)}{\sum_c \mathcal{N}(h_\phi(x_i); \mu_c, \sigma_c)}$
5. For each cluster c, compute mean $\mu_c = \frac{\sum_i z_{i,c} h_\phi(x_i)}{\sum_i z_{i,c}}$
6. Classify queries by Equation 6

Algorithm 1: Learning cluster variance

- If distance is small, closest instance dominates
- If distance is large, farther instances get involved

Learning cluster variance to better estimate the distance threshold (in the equation 5)

Actually, σ is differentiable so learn it from embedding



not an end-to-end model

Algorithm 2: Multi-modal clustering

- End-to-end optimization with non-differentiable λ
  - Soften the clustering
  - α as hyperparameter

- Find the best cluster for class “n”

$$c_n^* \leftarrow \arg\max_{c:l_c=n} \log p(h_\phi(x)|\mu_c, \sigma_c)$$

- Loss function

$$J = \frac{1}{|Q_n|} \sum_{x \in Q_n} \left[-\log p(h_\phi(x) \mid \mu_{c_n^*}, \sigma_{c_n^*}) + \log \sum_{n' \neq n} p(h_\phi(x) \mid \mu_{c_{n'}^*}, \sigma_{c_{n'}^*}) \right].$$

Result(1)

Table 1. Multi-modal clustering and learning cluster variances on fully-supervised 10-way, 10-shot Omniglot alphabet recognition and 5-way, 5-shot mini-ImageNet. Scaling distances with the learned variance gives a small improvement and multi-modal clustering gives a further improvement.

METHOD	σ	MULTI-MODAL	ALPH. ACC.	MINI. ACC.
PROTOTYPES	-	-	65.2 ± 0.6	66.1 ± 0.6
PROTOTYPES	✓	-	65.2 ± 0.6	67.2 ± 0.5
IMP (OURS)	✓	✓	$\mathbf{92.0} \pm 0.1$	$\mathbf{68.1} \pm 0.8$

Result(2)

Table 2. Learning labeled cluster variance σ_l and unlabeled cluster variance σ_u on semi-supervised 5-way, 1-shot Omniglot and mini-ImageNet with 5 unlabeled points per class and 5 distractors (see Section 4). Learning σ_l, σ_u is better than learning a tied σ for labeled and unlabeled clusters.

METHOD	σ	OMNI. ACC.	MINI. ACC.
TIED	σ	93.5±0.3	48.6±0.4
IMP (OURS)	σ_l, σ_u	**98.9**±0.1	**49.6**±0.8

Result(3)

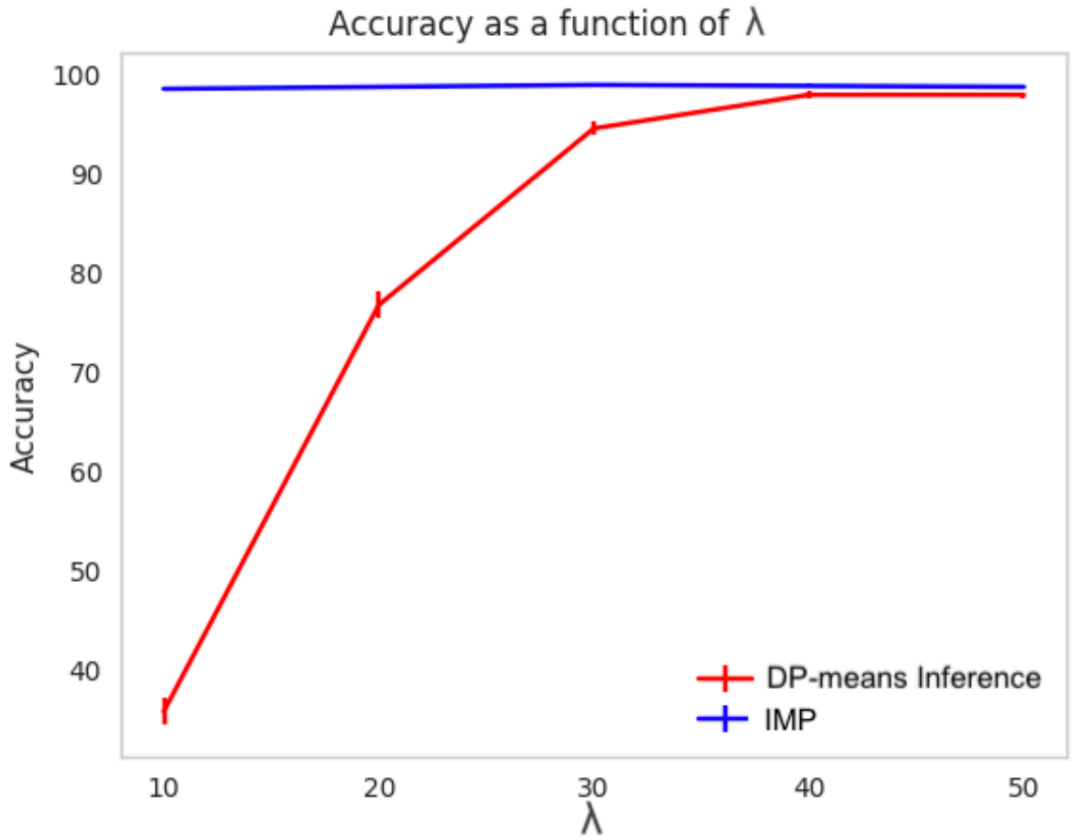


Figure 3. Learning and inference with IMP is more accurate and robust than DP-means inference on a prototypical network embedding alone. This plot shows the accuracy for the standard benchmark of semi-supervised 5-way, 1-shot Omniglot for different choices of the distance threshold λ for creating a new cluster.