GP-Tree: A Gaussian Process Classifier for Few-Shot Incremental Learning

Idan Achituve[1], Aviv Navon[1], Yochai Yemini[1], Gal Chechik[1,2], Ethan Fetaya[1]

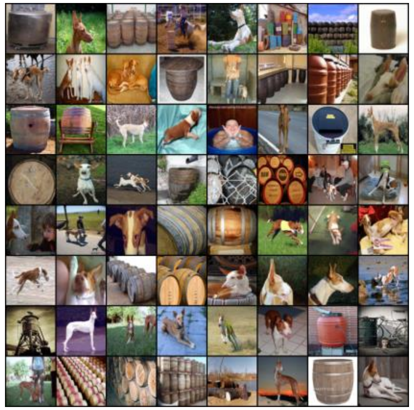
[1]Bar-Ilan University, Israel
[2]NVIDIA research, Israel

ICML 2021

Class-Incremental Learning

Train **Test**

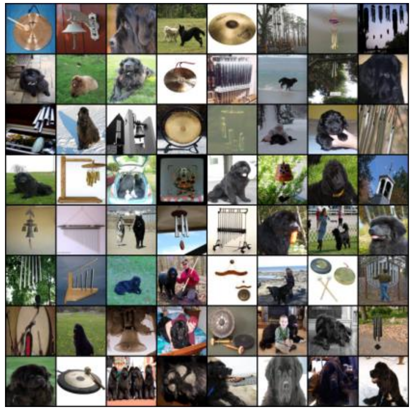
D_1

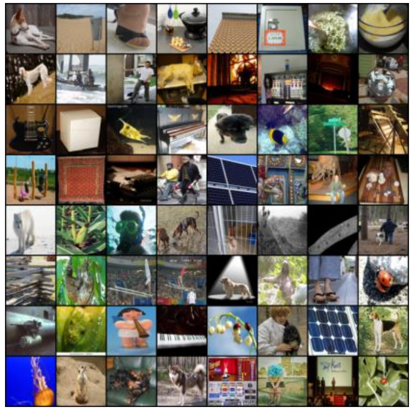


D_2

…

D_t





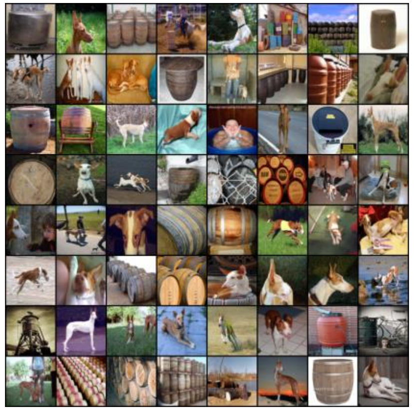
Base classes

Novel classes

Few-Shot Class-Incremental Learning (FSCIL)

Train **Test**

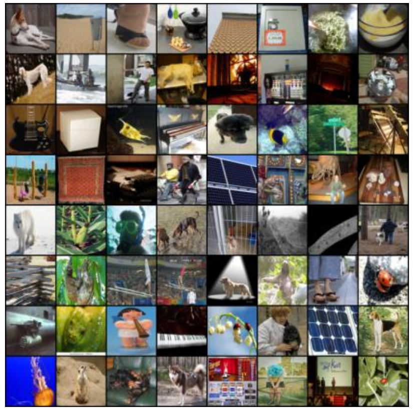
D_1



D_2

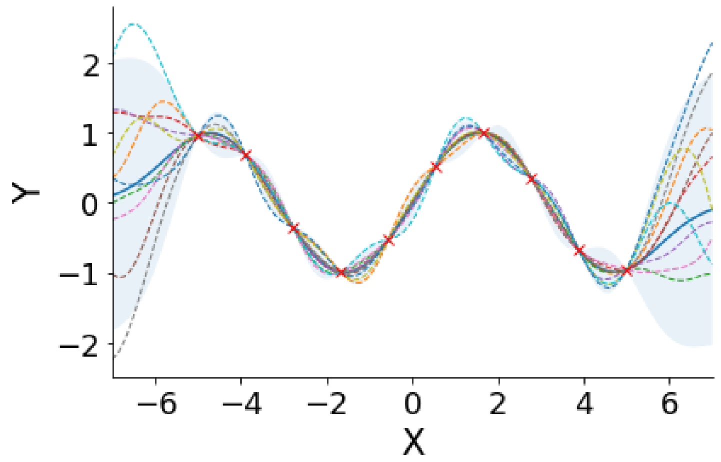
...

D_t



- The challenges here are two-fold:
  - Avoid catastrophic forgetting of previously encountered classes
  - Learning from few examples without overfitting

Gaussian Processes (GPs)



- GPs are Bayesian non-parametric models
- Controlled by a kernel function
- For regression, GP models enjoy closed-form marginal and posterior distributions
- Combined with neural networks via deep kernel learning (Wilson et al., 2016)
- Scale to large datasets with inducing points (Snelson & Ghahramani, 2006)

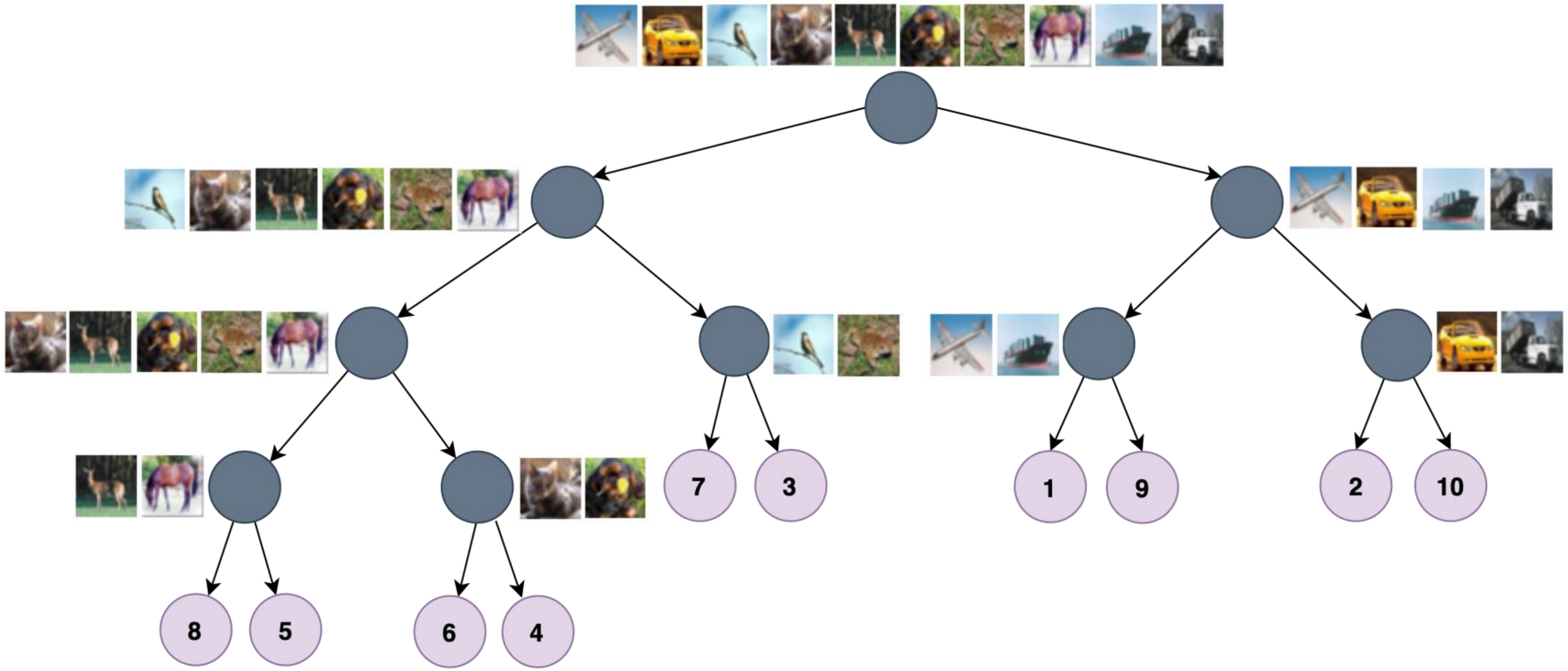
Why Gaussian Processes for FSCIL?

1. Store a compact summary of the base classes with inducing points methods
2. Generalize well from small datasets
3. Learn the kernel function parameters with the base classes dataset and avoid any parameter learning for novel classes

The Pólya-Gamma (PG) Augmentation (Polson et al., 2013)

- In classification the likelihood is not a Gaussian
- The Pólya-Gamma augmentation is a promising direction to overcome this limitation
- Suitable for binary classification tasks
- Current extension to multi-class classifications do not scale well with the number of classes

GP-Tree



Inference With GP-Tree

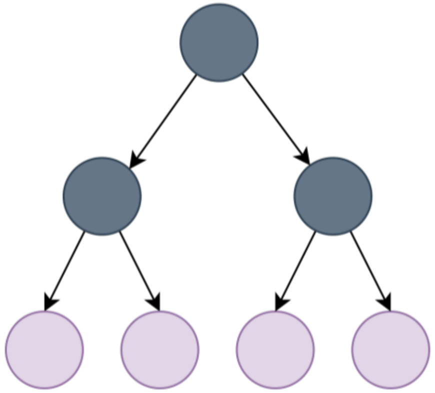
- Denote the GP associated with node v by $f_v \sim \mathcal{GP}(0, k_v(g_\theta(x), g_\theta(x')))$, all GPs in the tree by $\mathcal{F}$ and the neural network parameters by θ
- The likelihood,

$$p(y = c \,|\mathcal{F}) = \prod_{v \in P^c} \sigma(f_v)^{y_v} \big(1 - \sigma(f_v)\big)^{1 - y_v}$$

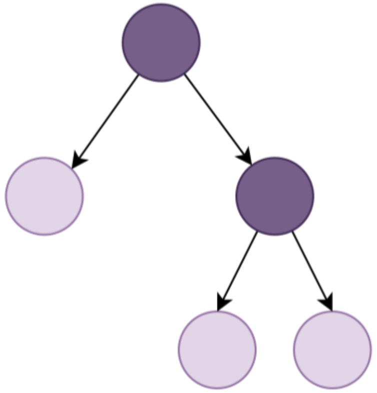
  where $y_v = 1$ if the path goes left at v and $y_v = 0$ otherwise
- Inference can be accomplished with either:
  - Variational inference
  - Gibbs sampling procedure

GP-Tree for Few-Shot Incremental Learning

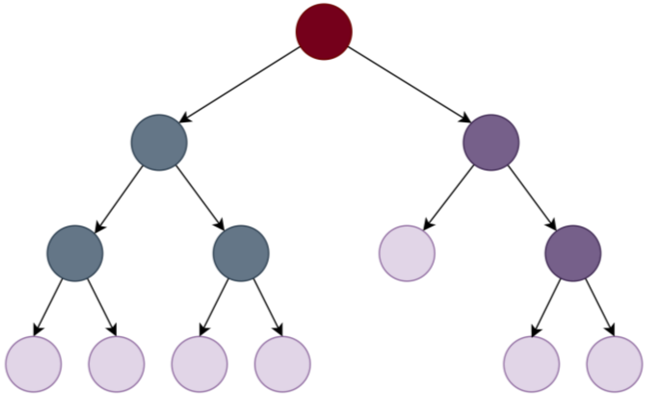
1. Build a tree from D_1



2. Build a sub-tree from $D_2, \dots, D_t$



3. Connect the trees with a shared root node



Classification on Few-Shot Class-Incremental Learning

Table 3. Few-shot class-incremental learning results on CUB-200-2011. Test accuracy averaged over 10 runs.

Method	Sessions										
	1	2	3	4	5	6	7	8	9	10	11
iCaRL	68.68	52.65	48.61	44.16	36.62	29.52	27.83	26.26	24.01	23.89	21.16
EEIL	68.68	53.63	47.91	44.20	36.30	27.46	25.93	24.70	23.95	24.13	22.11
NCM	68.68	57.12	44.21	28.78	26.71	25.66	24.62	21.52	20.12	20.06	19.87
TOPIC	68.68	62.49	54.81	49.99	45.25	41.40	38.35	35.36	32.22	28.31	26.28
SDC	64.10	60.58	57.00	53.57	52.09	49.87	48.20	46.38	44.04	43.81	42.39
PODNet	**75.93**	**70.29**	**64.50**	49.00	45.90	43.00	41.33	40.56	40.09	40.59	39.30
GP-Tree (ours)	73.73	68.24	64.22	**59.61**	**56.39**	**53.40**	**51.14**	**49.32**	**47.03**	**45.86**	**44.48**

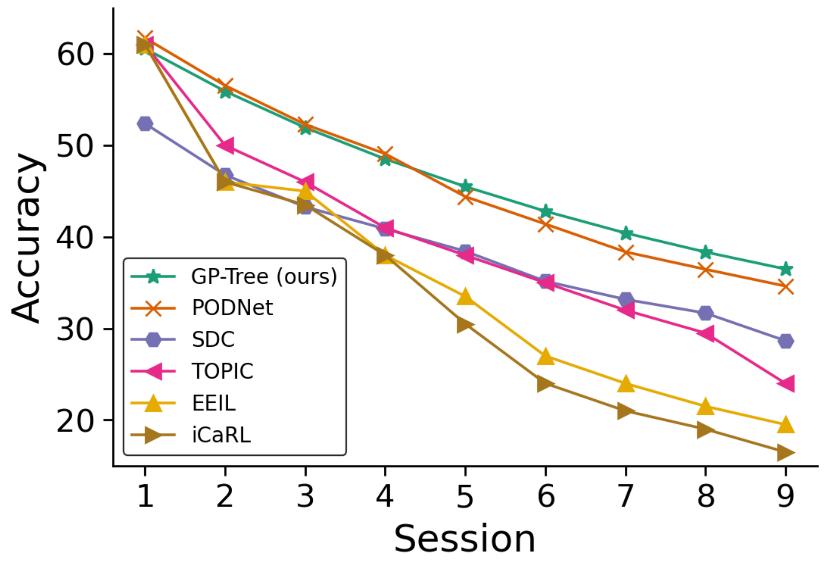


Figure 3. Class-incremental few-shot learning results on mini-ImageNet. Test set accuracy averaged over 10 runs.

Summary

- We presented GP-tree, a Gaussian process classifier that scales well with the number of classes and to large datasets
- Inference with GP-tree can be done either with a variational inference approach or a Gibbs sampling procedure
- GP-Tree can be easily adjusted to few-shot class-incremental learning challenges and achieves improved accuracy over current leading baseline methods

Thanks!

Code: https://github.com/IdanAchituve/GP-Tree