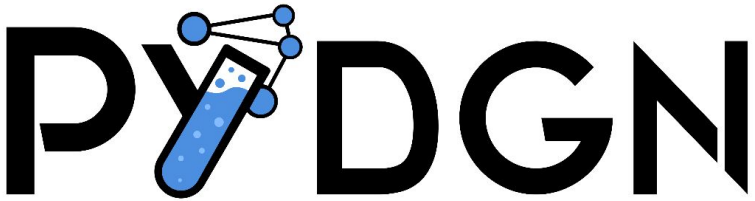


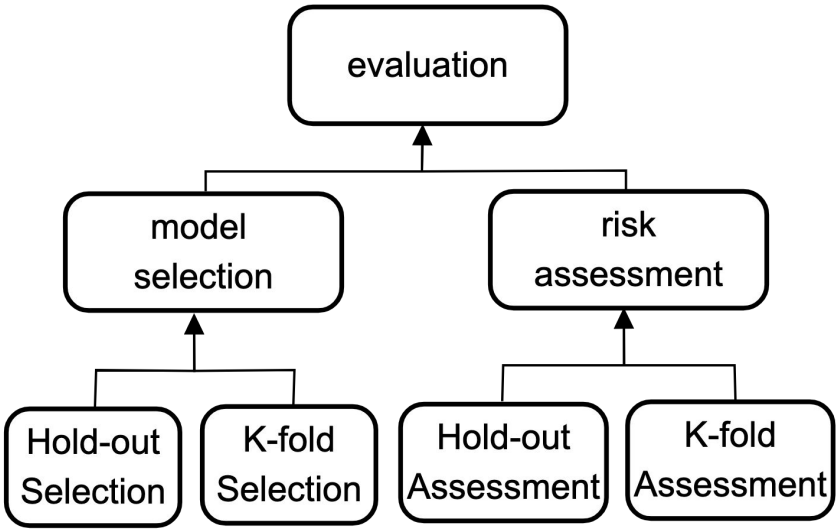
A Deep Graph Networks library for Python

https://github.com/diningphil/PyDGN

Federico Errica

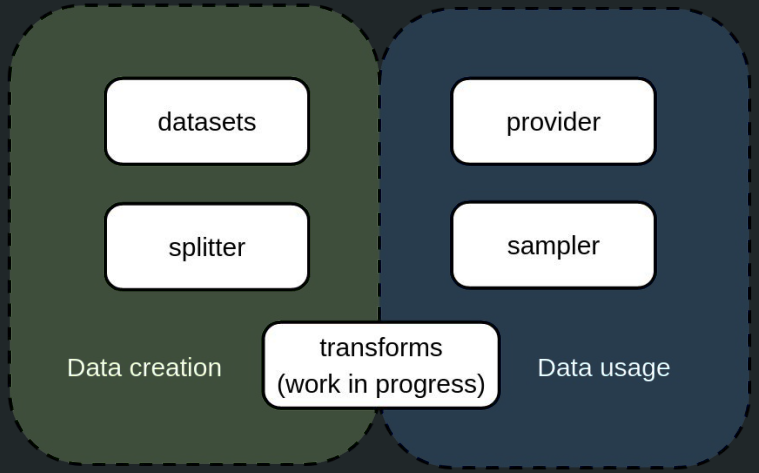
Robust Evaluation

(no more eval. mistakes)



Requirements:

- A dataset implementation
- A data split file

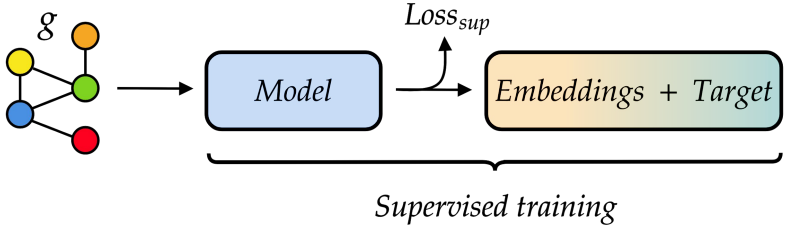


An Example

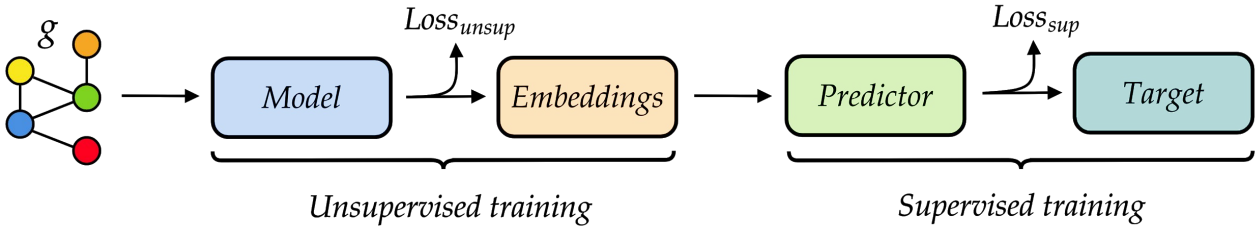
Data Preprocessing

```yaml
splitter:
  root: SPLITS/
  class_name: datasets.splitter.Splitter
  args:
    n_outer_folds: 10
    n_inner_folds: 1
    seed: 42
    stratify: True
    shuffle: True
    val_ratio: 0.1
    test_ratio: 0.1
dataset:
  root: DATA/
  class_name: datasets.datasets.TUDataset
  args:
    root: DATA/
    name: NCI1
  # useful for social datasets with no node features
  transforms:
      - class_name: datasets.transforms.ConstantIfEmpty
        args:
          value: 1
```

Three experiment classes (more are coming!)

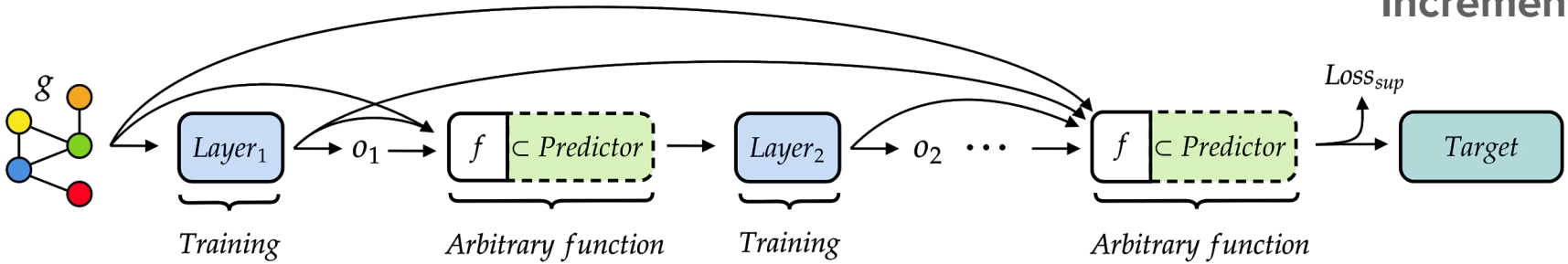


Supervised



Semi-supervised

Incremental

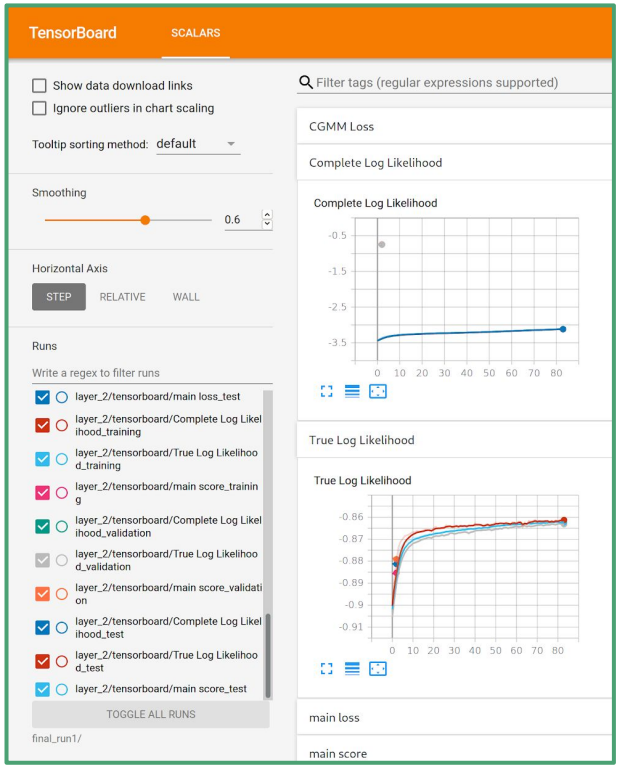


An Example

Supervised Experiment

```yaml
model: models.dgns.ToyDGN.ToyDGN
experiment: supervised
higher_results_are_better: True
log_every: 1
dataset-getter: datasets.provider.DataProvider
device: cpu
num_dataloader_workers: 0
pin_memory: False
grid:
  supervised_config:
    loss:
      - training.callback.loss.MulticlassClassificationLoss
    scorer:
      -
        class_name: training.callback.score.MultiScore
        args:
          # used at model selection time. Should be the one on which
          main_scorer: training.callback.score.MulticlassAccuracyScor
          my_metric2: training.callback.score.Toy1Score
    predictor:
      - models.predictors.GraphPredictor.LinearGraphPredictor
    early_stopper:
      -
        class_name:
          - training.callback.early_stopping.PatienceEarlyStopper
        args:
          patience:
            - 5
          # Should be the name of the main_scorer in MultiScore
          monitor: validation_Multiclass Accuracy # (train_,validation_)
          mode: max
          checkpoint: True
```

Plotting



Profiling

```
************************ Profiler ************************

Threshold: 1e-05

EngineCallback

   on_fetch_data --> Avg: 0.014047326353040013 s, Total: 1:15:44.928158
   on_forward --> Avg: 0.0027546234653820388 s, Total: 0:14:51.241894

MyLoss

   on_backward --> Avg: 0.0018598404143548941 s, Total: 0:04:44.109222
   on_training_batch_end --> Avg: 0.0005366264010137462 s, Total: 0:01:21.975049
```

Progress

```
******************** Experiment Progress ********************

 Out_1/Inn_1:  100%|#####|1/1, (1 cfg every 12:35:01.717716)
 Final run 1:  100%|#####|1/1, (1 cfg every 12:47:07.890757)
```

PyDGN final remarks

- Leverages Pytorch Geometric
  - Data processing
  - Graph convolution
- An extended library w.r.t. our ICLR paper
- In continuous development
  - Every project requires some improvement
- **Latest News:** Ray support for distributed computing
  - **by A.Carta (thanks!)**