Machine Learning Homework 2

109062131 林劭軒

1 Attributes

- `n_trees`: 100
- `n_features`: 16
- `sample_size`: 0.6
- `max_depth`: 3

With parallel processing, I can train 100 trees with depth 7 in 13 minutes.

2 Difficulty Encountered

The training is too slow for us to find an optimal set of attributes in short time.

3 Solutions and Reflections

I first used faster method to find best split such as let threshold be [`0.1`, `0.9`] with `step` = 0.1 quantile of the data instead of all data. After finding a better attribute, we can switch back to the full version.

After TA announced that we can use multiprocessing, I modified into the multiprocessing version. Since we can't use multiprocessing in jupyter notebook, I used multiprocess instead. Here is my implementation.

```
import multiprocess as mp
def build_forest(data: pd.DataFrame, n_trees, n_features,
    n_samples):
  x_features = list(data.columns)
  x_features.remove(Y_FEATURE)
  indices = list(data.index)
  forest = []
  def build_partial_tree(idx):
    part_features = random.sample(x_features, n_features)  # Get
        random features from all x features
    part_features.append(Y_FEATURE)
    part_indices = random.sample(indices, n_samples)  # Get random
         indices
    tree = build_tree(data[part_features].loc[part_indices],
        max_depth, min_samples_split, 0)  # Reusing build_tree
        function
    return tree
  with mp.Pool(mp.cpu_count()) as p:
    forest = p.map(build_partial_tree, range(n_trees))
  return forest
```