Stanford University

CS 329P : Practical Machine Learning (2021 Fall)

5.2 Bagging

Qingqing Huang, Mu Li, Alex Smola

https://c.d2l.ai/stanford-cs329p

Bagging - Bootstrap AGGregatING

- Learn n base learners in **parallel**, combine to reduce model variance
- Each base learner is trained on a bootstrap sample
  - Given a dataset of m examples, create a sample by randomly sampling m examples with replacement
  - Around $1 - 1/e \approx 63\,\%$ unique examples will be sampled use the out-of-bag examples for validation
- Combine learners by averaging the outputs (regression) or majority voting (classification)
- Random forest: bagging with decision trees
  - usually select random subset of features for each bootstrap sample

Bagging Code (scikit-learn)

```
class Bagging:
    def __init__(self, base_learner, n_learners):
        self.learners = [clone(base_learner) for _ in range(n_learners)]

    def fit(self, X, y):
        for learner in self.learners:
            examples = np.random.choice(
                np.arange(len(X)), int(len(X)), replace=True)
            learner.fit(X.iloc[examples, :], y.iloc[examples])

    def predict(self, X):
        preds = [learner.predict(X) for learner in self.learners]
        return np.array(preds).mean(axis=0)
```

Apply bagging with unstable Learners

- Bagging reduces model variance, especially for unstable learners
- Given ground truth f and a set of base learners $\hat{f}_D$ for regression, bagging prediction $\hat{f}(x) = \mathrm{E}_D[\hat{f}_D(x)]$
- $\mathrm{E}[X]^2 \leq \mathrm{E}[X^2]$

$$\underbrace{\left(f(x) - \hat{f}(x)\right)^2}_{\text{With bagging}} \leq \mathrm{E}\left[\underbrace{(f(x) - \hat{f}_D(x))^2}_{\text{Single learner}}\right] \Leftrightarrow \hat{f}(x)^2 \leq \mathrm{E}\left[\hat{f}_D(x)^2\right]$$

Unstable Learners

- Decision tree is unstable, linear regression is stable