

png
append (중복안되게 하면 좋음)

데이터 정리 (이상치 제거) → crop (중정) → 데이터 정리 (이상치 제거) → augmentation (공정) → Backbone.

image로 학습.

+ torch log 모든 것들을 기록 저장하는 것

→ 원본 및 안나들게 이미지 증강해서 넣는게 좋을 것 같아

→ Backbone Model ensemble

Ensemble

1. Bagging (Bootstrap Aggregating)

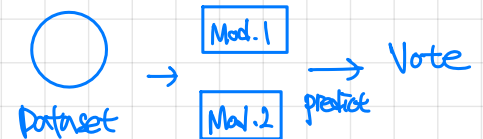
Bootstrap - random sampling (전체집합에서 무작위 복원추출을 통해 여러 복본집합 만들기) (중복 허용)

Aggregating - 집계 (평균이나 최빈값 도출)

→ 높은 bias의 underfitting or 높은 variance로 인한 overfitting 해결 도움.

categorical Data (classification) : 다수표 투표 결과 집계

Continuous Data (Regression) : 평균값 집계

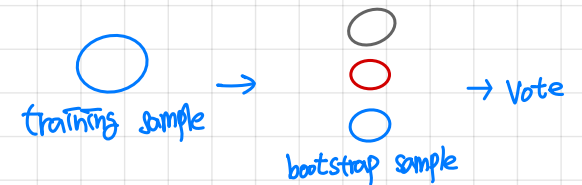


2. Voting (여러개의 Classifier가 Vote를 결정)

비슷 서로 다른 알고리즘(classifier)을 여러개 포함하여 사용)

Soft-voting : 확률 값을 평균 내서 확률이 가장 높은 레이블을 결정.

Hard-voting : 각 classifier가 예측한 레이블(label)을 투표해서 가장 투표수가 많은 레이블 결정



3. Boosting - 여러개의 분류기가 순차적으로 학습 수행

→ 이전 분류기가 예측이 틀린 데이터에 대해 올바르게 예측할 수 있도록 다음 분류기에 가중치 부여하여 학습, 예측 진행.

+ 예측 성능 뛰어나

- 학습이 느리고 과적합 발생 가능성 존재.

torchensemble

VotingClassifier (다양한 앙상블 모델 결합하여 다수표 투표에 기반한 예측 수행)

```
ensemble = VotingClassifier(estimators=[model1, model2, model3], voting='hard')
```

BaggingClassifier (샘플을 추출하여 개별 모델을 학습하고, 다수표 투표를 통해 예측 수행)
+ 각 모델은 독립적으로 학습.

```
ensemble = BaggingClassifier(estimators=[model1, model2, model3], n_estimators=3)
```

→ 앙상블에 포함될 개별 모델 개수

Snapshot Ensemble (여러개의 스냅샷 모델을 저장하고 앙상블) (특정시점의 모델 가중치 저장)

```
ensemble = SnapshotEnsembleClassifier(estimator=base_model, n_estimators=5, cuda=True)
```

Sklearn Ensemble

Gradient Boosting Classifier (이전 모델의 약점을 보완하는 방식, 순차적으로 학습)