

Python으로 머신러닝 입문



세션장: 구은아, 김은기

결정 트리

결정 트리 클래스

`sklearn.tree.DecisionTreeClassifier`

```
class sklearn.tree.DecisionTreeClassifier(*, criterion='gini', splitter='best', max_depth=None, min_samples_split=2, min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_features=None, random_state=None, max_leaf_nodes=None, min_impurity_decrease=0.0, class_weight=None, ccp_alpha=0.0)
```

[\[source\]](#)

<https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html>

`sklearn.tree.DecisionTreeRegressor`

```
class sklearn.tree.DecisionTreeRegressor(*, criterion='squared_error', splitter='best', max_depth=None, min_samples_split=2, min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_features=None, random_state=None, max_leaf_nodes=None, min_impurity_decrease=0.0, ccp_alpha=0.0)
```

[\[source\]](#)

<https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeRegressor.html>

결정 트리 파라미터

| 파라미터 | 설명 |
|------------------|--|
| min_sample_split | 노드를 분할하기 위한 최소한의 샘플 데이터 수 디폴트는 2 |
| min_sample_leaf | leaf 노드가 되기 위한 최소한의 샘플 데이터 수 |
| max_features | 분할에서 고려할 최대 피처 개수(int일 경우 개수, float일 경우 비율) 디폴트는 None |
| max_depth | 트리의 최대 깊이 디폴트는 None |
| max_leaf_nodes | leaf 노드의 최대 개수 |

보팅 분류기

보팅 분류기 클래스

`sklearn.ensemble.VotingClassifier`

```
class sklearn.ensemble.VotingClassifier(estimators, *, voting='hard', weights=None, n_jobs=None, flatten_transform=True, verbose=False)
```

[\[source\]](#)

<https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.VotingClassifier.html>

랜덤 포레스트

랜덤 포레스트 클래스

`sklearn.ensemble.RandomForestClassifier`

```
class sklearn.ensemble.RandomForestClassifier(n_estimators=100, *, criterion='gini', max_depth=None, min_samples_split=2, min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_features='auto', max_leaf_nodes=None, min_impurity_decrease=0.0, bootstrap=True, oob_score=False, n_jobs=None, random_state=None, verbose=0, warm_start=False, class_weight=None, ccp_alpha=0.0, max_samples=None)
```

[\[source\]](#)

<https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html>

`sklearn.ensemble.RandomForestRegressor`

```
class sklearn.ensemble.RandomForestRegressor(n_estimators=100, *, criterion='squared_error', max_depth=None, min_samples_split=2, min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_features='auto', max_leaf_nodes=None, min_impurity_decrease=0.0, bootstrap=True, oob_score=False, n_jobs=None, random_state=None, verbose=0, warm_start=False, ccp_alpha=0.0, max_samples=None)
```

[\[source\]](#)

<https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html>

랜덤 포레스트 파라미터

| 파라미터 | 설명 |
|------------------|--|
| n_estimator | 결정 트리의 개수 디폴트는 10 |
| min_sample_split | 노드를 분할하기 위한 최소한의 샘플 데이터 수 디폴트는 2 |
| min_sample_leaf | leaf 노드가 되기 위한 최소한의 샘플 데이터 수 |
| max_features | 분할에서 고려할 최대 피처 개수(int일 경우 개수, float일 경우 비율) 디폴트는 auto |
| max_depth | 트리의 최대 깊이 디폴트는 None |
| max_leaf_nodes | leaf 노드의 최대 개수 |

과제 안내

과제: 사용자 행동 인식 분류

main ▾ 21-2.ML-tutorial-with-python / 학습자료 / 3주차 분류1 /

eunai9 Create 3주차 과제 데이터

| | |
|--------------|-------------------|
| 3주차 과제 데이터 | Create 3주차 과 |
| 3주차_과제.ipynb | Add files via upl |
| t | Create t |

3주차_과제.ipynb 파일에 human activity 데이터를 전처리하여 train set와 test set로 분할까지 되어 있습니다.

- 1) human activity 데이터를 결정 트리로 분류해보고, feature_importances_ 메서드를 사용하여 어떤 피처가 높은 중요도를 갖고 있는지 파악해봅시다.
- 2) human activity 데이터를 랜덤 포레스트로 분류해보고, feature_importances_ 메서드를 사용하여 어떤 피처가 높은 중요도를 갖고 있는지 파악해봅시다.

decisiontreeclassifier와 randomforestclassifier의 파라미터도 변경해보면서 다양하게 분류기를 만들어보세요!

수고하셨습니다!
과제 열심히 하시고 다음 주에 보어요~