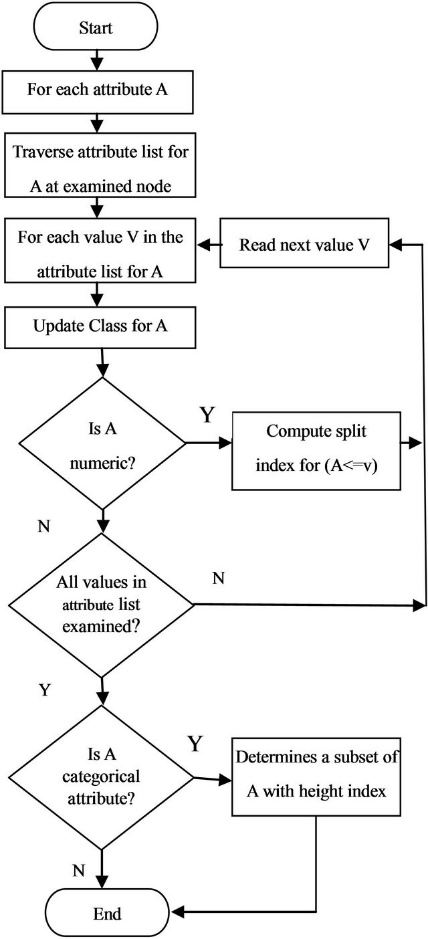
<결정 학습 트리 구현 및 실험>

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **과제명** | 결정 트리 학습 구현 및 실험 | | | | |
| **과목명** | 인공지능(H030-3-1196-01) | | | | |
| **성명** | 정승훈 | | **연락처** | **핸드폰** | 010-8648-7561 |
| **학과** | 전자통신공학과 | | **이메일** | tiktaktok116@naver.com |
| **학번** | 2015707003 | | **지도교수** | 박병준 교수님 | |
| **과제기간** | 2019.11.06 ~ 2019.11.20 | | | | |
| **개발 환경** | Operating System | Linux (Ubuntu 18.04) | | | |
| Programming Language | Python 3.6 | | | |
| IDLE | Jupyter Notebook | | | |
| Library | Scikit-learn, numpy, Image, pydotplus | | | |
| **과제**  **요구사항** | * Scikit-learn을 활용하여Dataset으로부터 결정 트리 생성하고 시각화 패키지를 이용하여 트리를 시각화하여 출력 * 최소 2개의 Dataset을 이용하여 실험 결과를 얻을 수 있게 구현   (1개는 Trainning Set, 나머지 1개는 Testing Set)   * 트리 생성 함수(DecisionTreeClassifier())에 대해 설정하는 파라미터들 (criterion(엔트로피 또는 지니계수), max\_depth(트리의 최대 깊이)등)을 달리하여 학습된 트리의 분류 정확도 출력 | | | | |
| **구현 내용** | * Scikit-learn 을 활용하여 Dataset으로부터 결정 트리 생성 * 시각화 패키지를 이용하여 트리를 시각화하여 출력 * scikit-learn에서 제공하는 iris, cancer 2개의 open Dataset을 이용하여 실험 결과를 얻을 수 있게 구현 * 트리 생성 함수(DecisionTreeClassifier())에 대해 설정하는 파라미터들 (criterion(엔트로피 또는 지니계수), max\_depth(트리의 최대 깊이)등)을 달리하여 학습된 트리의 분류 정확도 출력 * prunning을 통한 Tree-Depth 조절 * 시각화한 트리를 image 파일로 따로 저장하는 기능 추가 | | | | |

# flow chart



# Program Architecture

1. Datasets : scikit-learn에서 제공하는 데이터셋 이용
2. DecisionTreeClassifier : decision tree Machine learning Model 이용

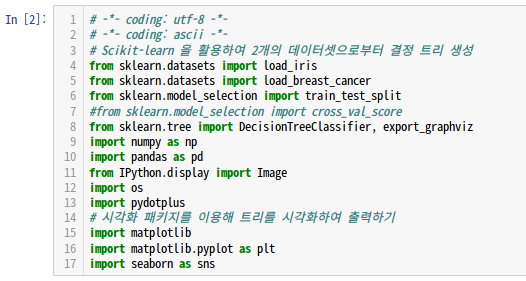
Parameter : Class\_weight=None, criterion, max\_depth, max\_features=None, max\_leaf\_nodes=None, min\_impurity\_decrease=0.0, min\_impurity\_split=None,

min\_samples\_leaf=1, min\_samples\_split=2, min\_weight\_fraction\_leaf=0.0, presort=False, random\_state=None, splitter='best')

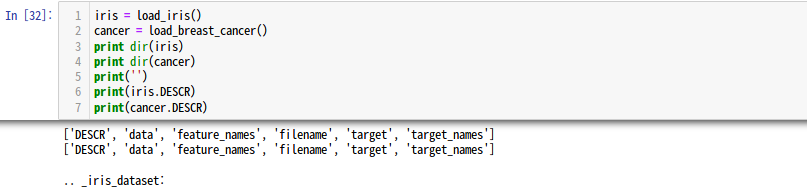
1. Tree.fit : train set을 맞추어 학습 모델을 만든다.
2. 분석 결과의 accuracy score 측정
3. Export\_graphviz : sklearn\_tree에서 제공하는 시각화 패키지 따로 모듈 설치도 진행한다.
4. 출력된 트리에서 직관적인 색에서 그룹을 분류한다.

# 구현 내용

1. Import module for Decision Tree Classification



1. Open dataset 정보 확인

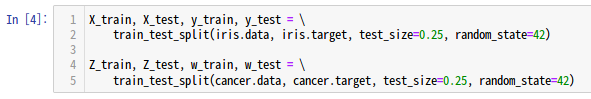


sklearn에서 가져온 오픈소스 데이터셋은 iris와 Cancer 두 개이다.

두 데이터셋 정보는 .ipynb에서 확인할 수 있다.

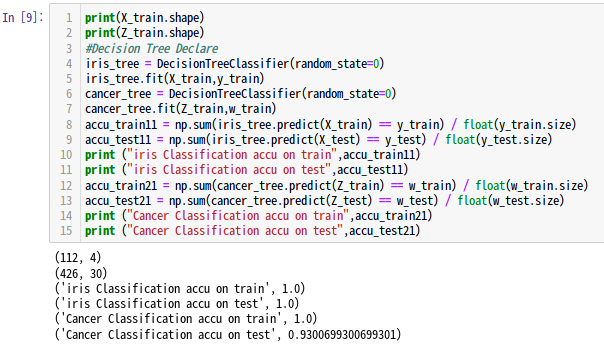
이 데이터셋에서 가져와 트리 정보에 넣을 내용은 class와 attribute이다.

1. Training set과 testing set 생성



Train\_test\_split 함수를 사용해 자동으로 데이터셋을 분리해준다.

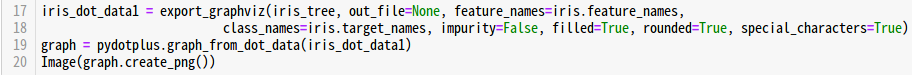
1. 기본 결정 트리 생성

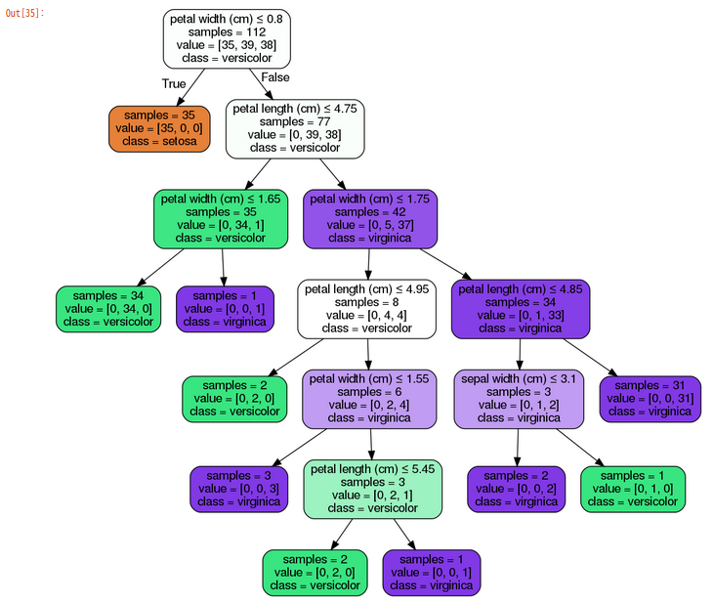


데이터셋의 정리가 완료되었으면 이제 위에서 연습한 방법과 동일한 방법으로 트리를 학습시킨다. 학습된 모델을 test 데이터로 테스트한 뒤, accuracy를 구하였다.

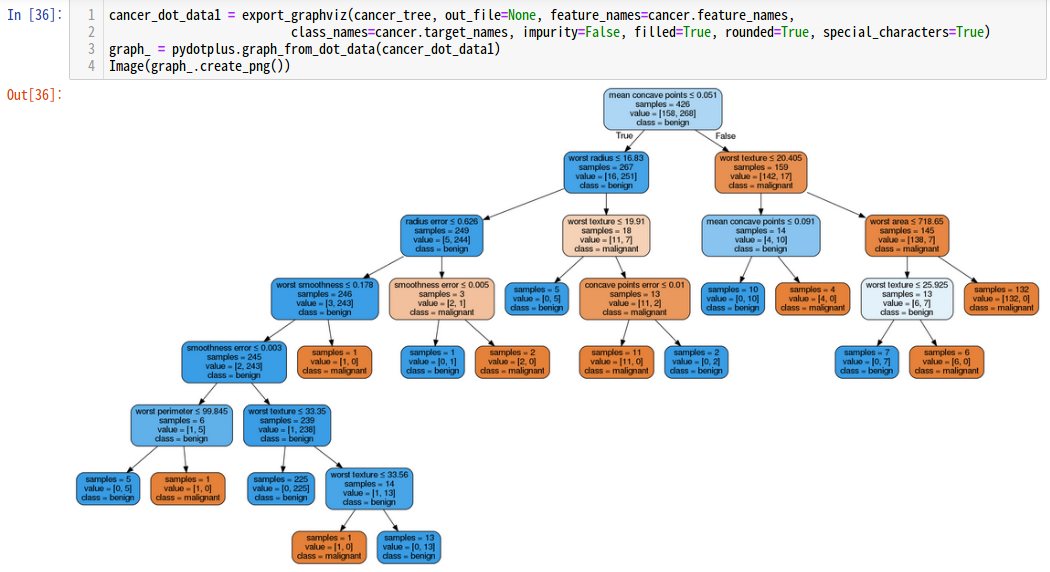
Tree.predict를 이용하여 overfit 현상을 방지하도록 설정한다.

DecisionTreeClassifier를 이용해 각각 데이터셋에 대한 트리를 생성하였다. 기준과 트리의 깊이를 설정하지 않고 먼저 결정 트리를 출력해보았다.





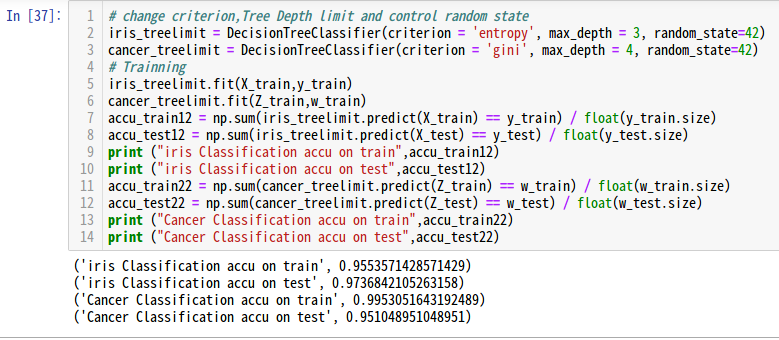
초기에 112개의 sample이 있었고 petal width가 0.8이하인 데이터들만 분류를 했을 때 모두 setsosa class였고 나머지 데이터들 중에 petal length가 4.75 이하인 데이터들만 분류했을 때 versicolor:virginica 비율이 34:1, 4.75 초과인 데이터들에서 versicolor:virginica 비율이4:17라는 것을 알 수 있다. 나머지 자식 노드들도 이와 같이 분류를 계속 진행하여 단말노드에 이른다. 더 이상 분류할 sample이 없을 때 자식노드를 생성하지 않는다. Virginica가 보라색 노드, versicolor가 초록색 노드, setosa class가 주황색 노드임을 알 수 있다.

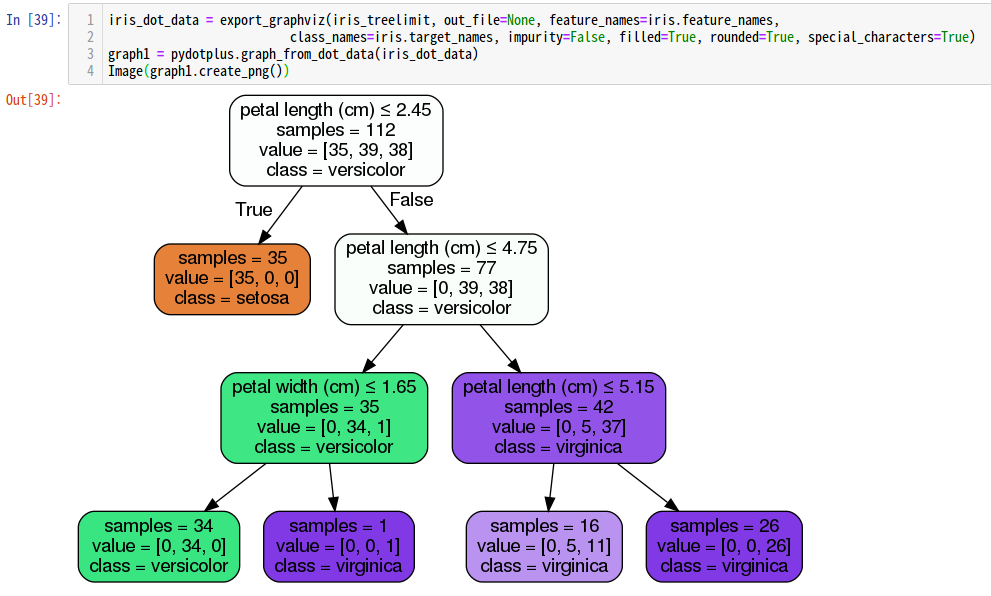


초기에 426개의 sample이 있고 mean concave points가 0.051이하 여부에 따라 데이터들을 분류를 한다. 분류했을 때 0.051 이하인 데이터들 중 class 비율은 benign : malignant = 249:18, 0.051 이상인 데이터들 중 class 비율은 benign : malignant = 14:145임을 알 수 있다. 이와 같은 방법으로 단말 노드까지 계속 분류하여 더 이상 분류할 sample이 없을 때 단말 노드에 이른다.

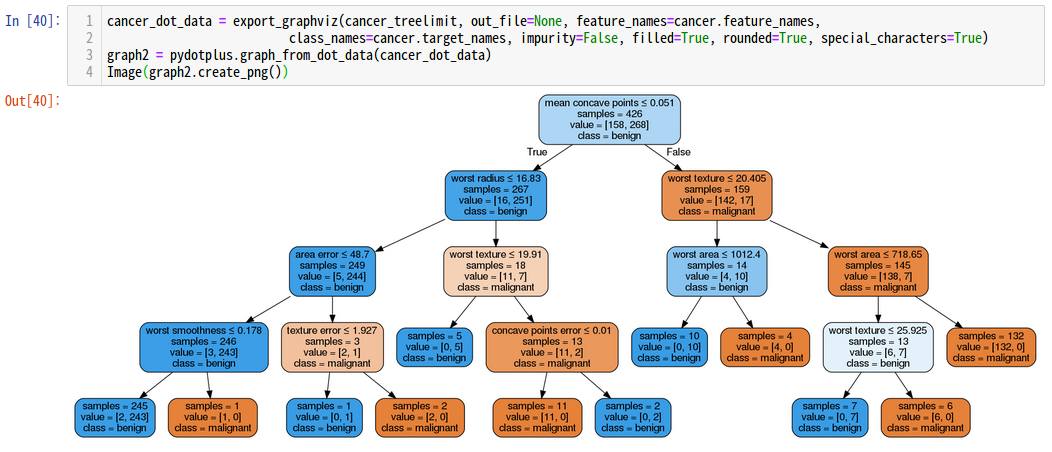
트리의 장점은 학습 모델을 시각화여 볼 수 있다는 장점이 있다. 하지만 파라미터를 설정하지 않았기 때문에 트리가 매우 복잡하므로 다시 설정하여 생성하는 것이 좋다.

1. 설정 값으로 결정 트리 재생성





불러온 Iris 데이터는 entropy를 기준으로 하고 트리의 최대 깊이를 3으로 설정, random\_state 값도 임의로 변경해서 출력해보았다. Entropy는 불순도를 측정하는 것이며 virginica class에서 pedal length와 value 기준이 변경되었는데 불순도가 감소했다는 것을 알 수 있고 이전에 기준을 적용하지 않았을 때 보다 정보를 얼마나 많이 얻었는지 알 수 있었다. 트리의 구성 내용은 이전에 파라미터를 적용하지 않았을 때 트리 내용과 같다.



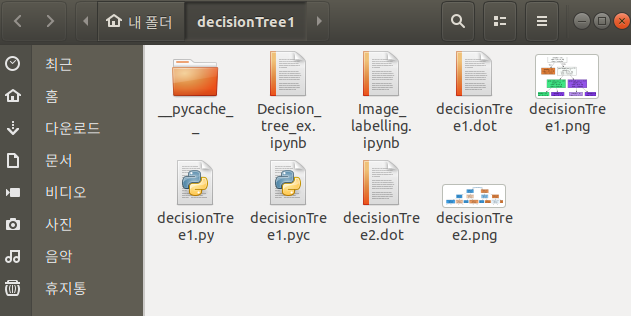
불러온 Cancer 데이터는 gini를 기준으로 하고 트리의 최대 깊이를 4로 설정, random\_state 값도 임의로 변경해서 출력해보았다. Gini index는 entropy 외에 불순도를 계산하기 위해 많이 사용되며 노드의 순도 즉, 관측치의 비율을 의미한다. 트리의 구성 내용은 이전에 파라미터를 적용하지 않았을 때 트리 내용과 같다.

트리를 Pruning 할 때 gini 지수, entropy 지수 중 하나를 사용할 수 있다

1. 트리 출력물 파일로 저장



출력 결과는 다음과 같다.



첨부한 동영상 파일에서 파일이 새로 생성됨을 확인할 수 있다.

# 참고문헌

1. <https://datascienceschool.net/view-notebook/577a01e24d4f456bb5060da6e47054e1/>
2. <https://datascienceschool.net/view-notebook/16c28c8c192147bfb3d4059474209e0a/>