**교육일지**

**교육 제목 : 머신러닝(tree)**

**교육 장소 : YGL C6 강의실**

**교육 일시 : 2021/10/20**

**#로지스틱 회귀로 와인분류**

**wine = pd.read\_csv('https://bit.ly/wine\_csv\_data')**

**wine.head()**

**wine.info()**

**wine.describe() #평균, 표준편차,최소, 최대값.**

**data = wine[['alcohol', 'sugar', 'pH']].to\_numpy()**

**target = wine['class'].to\_numpy()**

**from sklearn.model\_selection import train\_test\_split**

**train\_input, test\_input, train\_target, test\_target = train\_test\_split(**

**data, target, test\_size=0.2, random\_state=42)**

**print(train\_input.shape, test\_input.shape)**

**from sklearn.preprocessing import StandardScaler**

**ss = StandardScaler()**

**ss.fit(train\_input)**

**train\_scaled = ss.transform(train\_input)**

**test\_scaled = ss.transform(test\_input)**

**from sklearn.linear\_model import LogisticRegression**

**lr = LogisticRegression()**

**lr.fit(train\_scaled, train\_target)**

**print(lr.score(train\_scaled, train\_target))**

**print(lr.score(test\_scaled, test\_target))**

**결정트리**

**from sklearn.tree import DecisionTreeClassifier**

**dt = DecisionTreeClassifier(random\_state=42)**

**dt.fit(train\_scaled, train\_target)**

**print(dt.score(train\_scaled, train\_target)) #훈련세트**

**print(dt.score(test\_scaled, test\_target)) #테스트 세트**

**import matplotlib.pyplot as plt**

**from sklearn.tree import plot\_tree**

**plt.figure(figsize=(10,7))**

**plot\_tree(dt)**

**plt.show()**

**plt.figure(figsize=(10,7))**

**plot\_tree(dt, max\_depth=1, filled=True, feature\_names=['alcohol', 'sugar', 'pH'])**

**plt.show()**

**### 가지치기**

**dt = DecisionTreeClassifier(max\_depth=3, random\_state=42)**

**dt.fit(train\_scaled, train\_target)**

**print(dt.score(train\_scaled, train\_target))**

**print(dt.score(test\_scaled, test\_target))**

**plt.figure(figsize=(20,15))**

**plot\_tree(dt, filled=True, feature\_names=['alcohol', 'sugar', 'pH'])**

**plt.show()**

**dt = DecisionTreeClassifier(max\_depth=3, random\_state=42)**

**dt.fit(train\_input, train\_target)**

**print(dt.score(train\_input, train\_target))**

**print(dt.score(test\_input, test\_target))**

**plt.figure(figsize=(20,15))**

**plot\_tree(dt, filled=True, feature\_names=['alcohol', 'sugar', 'pH'])**

**plt.show()**

**print(dt.feature\_importances\_)**