Date: 25/5/2024

Decision tree ID3

```
import numpy as np
import pandas as pd
df = pd.read_csv('iris')
df.head()
df.info()
df.describe()
def find_entropy(df):
    target = df.keys()[-1]
    entropy = 0
    values = df[target].unique()
    for value in values:
        fraction = df[target].value_counts()[value] /
                      len(df[target])
        entropy += fraction * np.log2(fraction)
    return entropy
def buildTree(df, tree=None):
    target = df.keys()[-1]
    node = find_winner(df)
    att = np.unique(df[node])
    if tree is None:
        tree = {}
        tree[Node] = {}
    for value in att:
        sub = get_subtable(df, node, value)
        clValue, counts = np.unique(subtable[target],
                                     return_counts=True)

        if len(counts) == 1:
            tree[node][value] = clValue[0]
        else:
            tree[node][value] = buildTree(subtable)
    return tree
tree = buildTree(df)
import pprint
pprint.print(tree)
```

Decision tree (sklearn):

```
import pandas as pd
import numpy as np
import sklearn.model_selection import DecisionTree
from sklearn.tree import plot_tree

df = pd.read_csv('url')
dt.head()
df.info()
dt.isnull().sum()

cols = df.columns[0:-1]
for i in cols:
    sns.boxplot(y=dt[i])
    plt.show()

X = dt.drop('Species', axis=1)
y = df['Species']
X_train, X-test, y_train, y-test = train_test_split(X, y,
                                   test_size = 0,3)

dt = DecisionTreeClassifier(max_depth = 3)
dt.fit(x, y)

y_pred_train = dt.predict(x_train)
y-pred = dt.predict(x_test)

accuracy_score(y_pred, y_test)
```

9/5/24