

# ML Lab-1

Aim: To implement decision tree ML algorithm

Algorithm:

Decision tree algorithm has been applied on the IRIS dataset

The IRIS dataset is a classic dataset in machine learning that contains 150 samples of iris flowers, categorized into three species: Setosa, Versicolor, and Virginica. Each sample has four features: sepal length, sepal width, petal length, and petal width. The goal is to classify the species of iris based on these features

Step 1: Load the Iris dataset using `seaborn.load_dataset()` and separate features (x) and target (y)

Step 2: Split the dataset into training and testing sets using `train_test_split()`

Step 3: Initialize a Decision Tree classifier with the Gini index criterion.

Step 4: Train the model on the training data using `fit()` method.

Step 5: Make predictions on the test data using `predict()` method.

Step 6: Visualize the tree with `plot_tree()` and evaluate the model's accuracy using `accuracy_score()`

Code:

```
import pandas as pd
import seaborn as sns
from sklearn import tree

from sklearn.tree import DecisionTreeClassifier, plot_tree
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
import matplotlib.pyplot as plt

df=sns.load_dataset('iris')
df

x=df.drop(['species'], axis=1)
y=df.species
```

```
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3,random_state=55)
dt=DecisionTreeClassifier(criterion='gini')
dt.fit(x_train,y_train)
y_pred=dt.predict(x_test)

plt.figure(figsize=(10,20))
plot_tree(dt,feature_names=x.columns, class_names=y.unique(),filled=True)

accuracy_score(y_test,y_pred)*100
```

Output:

```
print(f"Accuracy = {accuracy_score(y_test,y_pred)*100}")
Accuracy = 95.55555555555556
```





