Sardar Patel Institute of Technology,Mumbai
Department of Electronics and Telecommunication Engineering
B.E. Sem-VII (2021-2022)
EC433 - Machine Learning and AI

Experiment: Decision Tree (ID3) algorithm

Name: Abhinav Pallayil **Date:28/09/2022**

Objective: Write Python program to demonstrate the working of the decision tree based ID3 algorithm by using appropriate data set for building the decision tree and apply this knowledge to classify a new sample.

Outcomes:

1. Find entropy of data and follow steps of the algorithm to construct a tree.
2. Representation of hypothesis using decision tree.
3. Apply Decision Tree algorithm to classify the given data.
4. Interpret the output of Decision Tree.

System Requirements:

Linux OS with Python and libraries or R or windows with MATLAB

Theory:

The decision tree builds classification or regression models in the form of a tree structure. It breaks down a dataset into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed. The final result is a tree with decision nodes and leaf nodes. A decision node (e.g., Outlook) has two or more branches (e.g., Sunny, Overcast and Rainy). Leaf node (e.g., Play) represents a classification or decision. The topmost decision node in a tree which corresponds to the best predictor called root node. Decision trees can handle both categorical and numerical data.

Entropy
A decision tree is built top-down from a root node and involves partitioning the data into subsets that contain instances with similar values (homogenous). ID3 algorithm uses entropy to calculate the homogeneity of a sample. If the sample is completely homogeneous the entropy is zero and if the sample is an equally divided it has entropy of one.

E(S) is the Entropy of the entire set, while the second term E(S, A) relates to an Entropy of an attribute A.

$$E(S) = \sum_{x \in X} -P(x) \log_2 P(x)$$

$$E(S, A) = \sum_{x \in X} [P(x) * E(S)]$$

Information Gain
The information gain is based on the decrease in entropy after a dataset is split on an attribute. Constructing a decision tree is all about finding attribute that returns the highest information gain (i.e., the most homogeneous branches).

$$IG(S, A) = E(S) - E(S, A)$$

Dataset:

This dataset describes whether a person will earn salary more than 100k based on company,job and details.

company	job	degree	salary_more_then_100k
google	sales executive	bachelors	No
google	sales executive	masters	No
google	business manager	bachelors	Yes
google	business manager	masters	Yes
google	computer programmer	bachelors	No
google	computer programmer	masters	Yes
abc pharma	sales executive	masters	No
abc pharma	computer programmer	bachelors	No
abc pharma	business manager	bachelors	No
abc pharma	business manager	masters	Yes
facebook	sales executive	bachelors	Yes
facebook	sales executive	masters	Yes
facebook	business manager	bachelors	Yes
facebook	business manager	masters	Yes
facebook	computer programmer	bachelors	Yes
facebook	computer programmer	masters	Yes

[5]:
```
import pandas as pd
import numpy as np
```

[8]:
```
df = pd.read_csv("/content/D.csv")
```

[9]:
```
df
```

[9]:

	company	job	degree	salary_more_then_100k
0	google	sales executive	bachelors	No
1	google	sales executive	masters	No
2	google	business manager	bachelors	Yes
3	google	business manager	masters	Yes
4	google	computer programmer	bachelors	No
5	google	computer programmer	masters	Yes
6	abc pharma	sales executive	masters	No
7	abc pharma	computer programmer	bachelors	No
8	abc pharma	business manager	bachelors	No
9	abc pharma	business manager	masters	Yes
10	facebook	sales executive	bachelors	Yes
11	facebook	sales executive	masters	Yes
12	facebook	business manager	bachelors	Yes
13	facebook	business manager	masters	Yes
14	facebook	computer programmer	bachelors	Yes
15	facebook	computer programmer	masters	Yes

[10]:
```
t = df.keys()[-1]
print('Target Attribute is    ', t)

# Get the attribute names from input dataset
attribute_names = list(df.keys())

#Remove the target attribute from the attribute names list
attribute_names.remove(t)

print('Predicting Attributes  ', attribute_names)
```

```
Target Attribute is     salary_more_then_100k
Predicting Attributes   ['company', 'job', 'degree']
```

[11]:
```
import math
def entropy(probs):
    return sum( [-prob*math.log(prob, 2) for prob in probs])

#Function to calulate the entropy of the given Datasets/List with respect to␣
 ↪target attributes
def entropy_of_list(ls,value):
    from collections import Counter

    # Total intances associated with respective attribute
    total_instances = len(ls)  # = 14
    print("-----------------------------------------------------------")
    print("\nTotal no of instances/records associated with '{0}' is  {1}".
 ↪format(value,total_instances))
    # Counter calculates the propotion of class
    cnt = Counter(x for x in ls)
    print('\nTarget attribute class count(Yes/No)=',dict(cnt))

    # x means no of YES/NO
    probs = [x / total_instances for x in cnt.values()]
    print("\nClasses", max(cnt), min(cnt))
    print("\nProbabilities of Class 'p'='{0}'  {1}".format(max(cnt),max(probs)))
    print("Probabilities of Class 'n'='{0}'   {1}".format(min(cnt),min(probs)))

    # Call Entropy
    return entropy(probs)
```

[12]:
```
def information_gain(df, split_attribute, target_attribute,battr):
    print("\n\n----- Information Gain Calculation of",split_attribute,"----- ")

    # group the data based on attribute values
    df_split = df.groupby(split_attribute)
    glist=[]
    for gname,group in df_split:
        print('Grouped Attribute Values \n',group)
        print("-----------------------------------------------------------")
        glist.append(gname)

    glist.reverse()
    nobs = len(df.index) * 1.0
    df_agg1=df_split.agg({target_attribute:lambda x:entropy_of_list(x, glist.
 ↪pop())})
    df_agg2=df_split.agg({target_attribute :lambda x:len(x)/nobs})

    df_agg1.columns=['Entropy']
    df_agg2.columns=['Proportion']
```

```
    # Calculate Information Gain:
    new_entropy = sum( df_agg1['Entropy'] * df_agg2['Proportion'])
    if battr !='S':
        old_entropy = entropy_of_list(df[target_attribute],'S-'+df.iloc[0][df.
↪columns.get_loc(battr)])
    else:
        old_entropy = entropy_of_list(df[target_attribute],battr)
    return old_entropy - new_entropy
```

[15]:
```
def id3(df, target_attribute, attribute_names,␣
↪default_class=None,default_attr='S'):

    from collections import Counter
    cnt = Counter(x for x in df[target_attribute])# class of YES /NO

    ## First check: Is this split of the dataset homogeneous?
    if len(cnt) == 1:
        return next(iter(cnt))  # next input data set, or raises StopIteration␣
↪when EOF is hit.

    ## Second check: Is this split of the dataset empty? if yes, return a␣
↪default value
    elif df.empty or (not attribute_names):
        return default_class  # Return None for Empty Data Set

    ## Otherwise: This dataset is ready to be devied up!
    else:
        # Get Default Value for next recursive call of this function:
        default_class = max(cnt.keys()) #No of YES and NO Class
        # Compute the Information Gain of the attributes:
        gainz=[]
        for attr in attribute_names:
            ig= information_gain(df, attr, target_attribute,default_attr)
            gainz.append(ig)
            print('\nInformation gain of','“',attr,'”','is ', ig)
            print("============================================================")

        index_of_max = gainz.index(max(gainz))               # Index of Best␣
↪Attribute
        best_attr = attribute_names[index_of_max]            # Choose Best␣
↪Attribute to split on
        print("\nList of Gain for arrtibutes:",attribute_names,"\nare:",␣
↪gainz,"respectively.")


        # Create an empty tree, to be populated in a moment
```

```
        tree = {best_attr:{}} # Initiate the tree with best attribute as a node
        remaining_attribute_names =[i for i in attribute_names if i != best_attr]

        # Split dataset-On each split, recursively call this algorithm.Populate␣
↪the empty tree with subtrees, which
        # are the result of the recursive call
        for attr_val, data_subset in df.groupby(best_attr):
            subtree = id3(data_subset,target_attribute,␣
↪remaining_attribute_names,default_class,best_attr)
            tree[best_attr][attr_val] = subtree
        return tree
```

[16]:
```
#Function to calulate the entropy of the given Dataset with respect to target␣
↪attributes
def entropy_dataset(a_list):
    from collections import Counter

    # Counter calculates the propotion of class
    cnt = Counter(x for x in a_list)
    num_instances = len(a_list)*1.0    # = 14
    print("\nNumber of Instances of the Current Sub-Class is {0}".
↪format(num_instances ))

    # x means no of YES/NO
    probs = [x / num_instances for x in cnt.values()]
    print("\nClasses", "'p'=",max(cnt), "'n'=",min(cnt))
    print("\nProbabilities of Class 'p'='{0}'  {1}".format(max(cnt),max(probs)))
    print("Probabilities of Class 'n'='{0}'   {1}".format(min(cnt),min(probs)))

    # Call Entropy
    return entropy(probs)

# The initial entropy of the YES/NO attribute for our dataset.
print("Entropy calculation for input dataset:\n")
print(df['salary_more_then_100k'])

total_entropy = entropy_dataset(df['salary_more_then_100k'])
print("\nTotal Entropy(S) of PlayGolf Dataset", total_entropy)
print("=========================================================")
#####################################################

from pprint import pprint
tree = id3(df,t,attribute_names)
print("\nThe Resultant Decision Tree is: \n")
pprint(tree)

attribute = next(iter(tree))
```

```
print("\nBest Attribute ",attribute)
print("Tree Keys      ",tree[attribute].keys())
```

```
Entropy calculation for input dataset:

0      No
1      No
2     Yes
3     Yes
4      No
5     Yes
6      No
7      No
8      No
9     Yes
10    Yes
11    Yes
12    Yes
13    Yes
14    Yes
15    Yes
Name: salary_more_then_100k, dtype: object

Number of Instances of the Current Sub-Class is 16.0

Classes 'p'= Yes 'n'= No

Probabilities of Class 'p'='Yes'  0.625
Probabilities of Class 'n'='No'   0.375

Total Entropy(S) of PlayGolf Dataset 0.9544340029249649
============================================================


----- Information Gain Calculation of company -----
Grouped Attribute Values
       company                  job     degree salary_more_then_100k
6  abc pharma      sales executive    masters                     No
7  abc pharma  computer programmer  bachelors                     No
8  abc pharma     business manager  bachelors                     No
9  abc pharma     business manager    masters                    Yes
------------------------------------------------------------
Grouped Attribute Values
      company                  job     degree salary_more_then_100k
10  facebook      sales executive  bachelors                    Yes
11  facebook      sales executive    masters                    Yes
12  facebook     business manager  bachelors                    Yes
13  facebook     business manager    masters                    Yes
```

```
14  facebook  computer programmer  bachelors                  Yes
15  facebook  computer programmer    masters                  Yes
-----------------------------------------------------------
Grouped Attribute Values
   company                  job     degree salary_more_then_100k
0  google      sales executive  bachelors                    No
1  google      sales executive    masters                    No
2  google     business manager  bachelors                   Yes
3  google     business manager    masters                   Yes
4  google  computer programmer  bachelors                    No
5  google  computer programmer    masters                   Yes
-----------------------------------------------------------
-----------------------------------------------------------

Total no of instances/records associated with 'abc pharma' is  4

Target attribute class count(Yes/No)= {'No': 3, 'Yes': 1}

Classes Yes No

Probabilities of Class 'p'='Yes'  0.75
Probabilities of Class 'n'='No'   0.25
-----------------------------------------------------------

Total no of instances/records associated with 'facebook' is  6

Target attribute class count(Yes/No)= {'Yes': 6}

Classes Yes Yes

Probabilities of Class 'p'='Yes'  1.0
Probabilities of Class 'n'='Yes'   1.0
-----------------------------------------------------------

Total no of instances/records associated with 'google' is  6

Target attribute class count(Yes/No)= {'No': 3, 'Yes': 3}

Classes Yes No

Probabilities of Class 'p'='Yes'  0.5
Probabilities of Class 'n'='No'   0.5
-----------------------------------------------------------

Total no of instances/records associated with 'S' is  16

Target attribute class count(Yes/No)= {'No': 6, 'Yes': 10}
```

```
Classes Yes No

Probabilities of Class 'p'='Yes'  0.625
Probabilities of Class 'n'='No'   0.375

Information gain of ‘‘ company ’’ is  0.3766144718101817
============================================================


----- Information Gain Calculation of job -----
Grouped Attribute Values
        company               job     degree salary_more_then_100k
2       google  business manager  bachelors                   Yes
3       google  business manager    masters                   Yes
8   abc pharma  business manager  bachelors                    No
9   abc pharma  business manager    masters                   Yes
12    facebook  business manager  bachelors                   Yes
13    facebook  business manager    masters                   Yes
------------------------------------------------------------
Grouped Attribute Values
        company                   job     degree salary_more_then_100k
4       google  computer programmer  bachelors                    No
5       google  computer programmer    masters                   Yes
7   abc pharma  computer programmer  bachelors                    No
14    facebook  computer programmer  bachelors                   Yes
15    facebook  computer programmer    masters                   Yes
------------------------------------------------------------
Grouped Attribute Values
        company              job     degree salary_more_then_100k
0       google  sales executive  bachelors                    No
1       google  sales executive    masters                    No
6   abc pharma  sales executive    masters                    No
10    facebook  sales executive  bachelors                   Yes
11    facebook  sales executive    masters                   Yes
------------------------------------------------------------
------------------------------------------------------------

Total no of instances/records associated with 'business manager' is  6

Target attribute class count(Yes/No)= {'Yes': 5, 'No': 1}

Classes Yes No

Probabilities of Class 'p'='Yes'  0.8333333333333334
Probabilities of Class 'n'='No'   0.16666666666666666
------------------------------------------------------------

Total no of instances/records associated with 'computer programmer' is  5
```

```
Target attribute class count(Yes/No)= {'No': 2, 'Yes': 3}

Classes Yes No

Probabilities of Class 'p'='Yes'  0.6
Probabilities of Class 'n'='No'   0.4
-----------------------------------------------------------

Total no of instances/records associated with 'sales executive' is  5

Target attribute class count(Yes/No)= {'No': 3, 'Yes': 2}

Classes Yes No

Probabilities of Class 'p'='Yes'  0.6
Probabilities of Class 'n'='No'   0.4
-----------------------------------------------------------

Total no of instances/records associated with 'S' is  16

Target attribute class count(Yes/No)= {'No': 6, 'Yes': 10}

Classes Yes No

Probabilities of Class 'p'='Yes'  0.625
Probabilities of Class 'n'='No'   0.375

Information gain of ‘‘ job ’’ is  0.10383147327266418
============================================================


----- Information Gain Calculation of degree -----
Grouped Attribute Values
        company                  job     degree salary_more_then_100k
0       google      sales executive  bachelors                     No
2       google     business manager  bachelors                    Yes
4       google  computer programmer  bachelors                     No
7   abc pharma  computer programmer  bachelors                     No
8   abc pharma     business manager  bachelors                     No
10    facebook      sales executive  bachelors                    Yes
12    facebook     business manager  bachelors                    Yes
14    facebook  computer programmer  bachelors                    Yes
-----------------------------------------------------------
Grouped Attribute Values
        company                  job   degree salary_more_then_100k
1       google      sales executive  masters                     No
3       google     business manager  masters                    Yes
```

```
5       google  computer programmer  masters                  Yes
6   abc pharma       sales executive  masters                   No
9   abc pharma     business manager  masters                  Yes
11    facebook       sales executive  masters                  Yes
13    facebook     business manager  masters                  Yes
15    facebook  computer programmer  masters                  Yes
-----------------------------------------------------------
-----------------------------------------------------------

Total no of instances/records associated with 'bachelors' is  8

Target attribute class count(Yes/No)= {'No': 4, 'Yes': 4}

Classes Yes No

Probabilities of Class 'p'='Yes'  0.5
Probabilities of Class 'n'='No'   0.5
-----------------------------------------------------------

Total no of instances/records associated with 'masters' is  8

Target attribute class count(Yes/No)= {'No': 2, 'Yes': 6}

Classes Yes No

Probabilities of Class 'p'='Yes'  0.75
Probabilities of Class 'n'='No'   0.25
-----------------------------------------------------------

Total no of instances/records associated with 'S' is  16

Target attribute class count(Yes/No)= {'No': 6, 'Yes': 10}

Classes Yes No

Probabilities of Class 'p'='Yes'  0.625
Probabilities of Class 'n'='No'   0.375

Information gain of ‘‘ degree ’’ is  0.04879494069539847
===========================================================

List of Gain for arrtibutes: ['company', 'job', 'degree']
are: [0.3766144718101817, 0.10383147327266418, 0.04879494069539847]
respectively.


----- Information Gain Calculation of job -----
Grouped Attribute Values
```

```
       company               job     degree salary_more_then_100k
8  abc pharma  business manager  bachelors                    No
9  abc pharma  business manager    masters                   Yes
-----------------------------------------------------------
Grouped Attribute Values
       company                  job     degree salary_more_then_100k
7  abc pharma  computer programmer  bachelors                    No
-----------------------------------------------------------
Grouped Attribute Values
       company              job   degree salary_more_then_100k
6  abc pharma  sales executive  masters                    No
-----------------------------------------------------------
-----------------------------------------------------------

Total no of instances/records associated with 'business manager' is  2

Target attribute class count(Yes/No)= {'No': 1, 'Yes': 1}

Classes Yes No

Probabilities of Class 'p'='Yes'  0.5
Probabilities of Class 'n'='No'   0.5
-----------------------------------------------------------

Total no of instances/records associated with 'computer programmer' is  1

Target attribute class count(Yes/No)= {'No': 1}

Classes No No

Probabilities of Class 'p'='No'  1.0
Probabilities of Class 'n'='No'   1.0
-----------------------------------------------------------

Total no of instances/records associated with 'sales executive' is  1

Target attribute class count(Yes/No)= {'No': 1}

Classes No No

Probabilities of Class 'p'='No'  1.0
Probabilities of Class 'n'='No'   1.0
-----------------------------------------------------------

Total no of instances/records associated with 'S-abc pharma' is  4

Target attribute class count(Yes/No)= {'No': 3, 'Yes': 1}
```

```
Classes Yes No

Probabilities of Class 'p'='Yes'  0.75
Probabilities of Class 'n'='No'   0.25

Information gain of “ job ” is  0.31127812445913283
============================================================


----- Information Gain Calculation of degree -----
Grouped Attribute Values
       company                  job     degree salary_more_then_100k
7  abc pharma  computer programmer  bachelors                     No
8  abc pharma     business manager  bachelors                     No
------------------------------------------------------------
Grouped Attribute Values
       company               job   degree salary_more_then_100k
6  abc pharma   sales executive  masters                     No
9  abc pharma  business manager  masters                    Yes
------------------------------------------------------------
------------------------------------------------------------

Total no of instances/records associated with 'bachelors' is  2

Target attribute class count(Yes/No)= {'No': 2}

Classes No No

Probabilities of Class 'p'='No'  1.0
Probabilities of Class 'n'='No'   1.0
------------------------------------------------------------

Total no of instances/records associated with 'masters' is  2

Target attribute class count(Yes/No)= {'No': 1, 'Yes': 1}

Classes Yes No

Probabilities of Class 'p'='Yes'  0.5
Probabilities of Class 'n'='No'   0.5
------------------------------------------------------------

Total no of instances/records associated with 'S-abc pharma' is  4

Target attribute class count(Yes/No)= {'No': 3, 'Yes': 1}

Classes Yes No
```

```
Probabilities of Class 'p'='Yes'  0.75
Probabilities of Class 'n'='No'   0.25

Information gain of ‘‘ degree ’’ is  0.31127812445913283
============================================================

List of Gain for arrtibutes: ['job', 'degree']
are: [0.31127812445913283, 0.31127812445913283] respectively.


----- Information Gain Calculation of degree -----
Grouped Attribute Values
       company               job     degree salary_more_then_100k
8  abc pharma  business manager  bachelors                     No
-------------------------------------------------------------
Grouped Attribute Values
       company               job   degree salary_more_then_100k
9  abc pharma  business manager  masters                    Yes
-------------------------------------------------------------
-------------------------------------------------------------

Total no of instances/records associated with 'bachelors' is  1

Target attribute class count(Yes/No)= {'No': 1}

Classes No No

Probabilities of Class 'p'='No'  1.0
Probabilities of Class 'n'='No'   1.0
-------------------------------------------------------------

Total no of instances/records associated with 'masters' is  1

Target attribute class count(Yes/No)= {'Yes': 1}

Classes Yes Yes

Probabilities of Class 'p'='Yes'  1.0
Probabilities of Class 'n'='Yes'   1.0
-------------------------------------------------------------

Total no of instances/records associated with 'S-business manager' is  2

Target attribute class count(Yes/No)= {'No': 1, 'Yes': 1}

Classes Yes No

Probabilities of Class 'p'='Yes'  0.5
```

```
Probabilities of Class 'n'='No'   0.5

Information gain of ‘‘ degree ’’ is  1.0
=========================================================

List of Gain for arrtibutes: ['degree']
are: [1.0] respectively.


----- Information Gain Calculation of job -----
Grouped Attribute Values
   company               job     degree salary_more_then_100k
2  google  business manager  bachelors                   Yes
3  google  business manager    masters                   Yes
---------------------------------------------------------
Grouped Attribute Values
   company                  job     degree salary_more_then_100k
4  google  computer programmer  bachelors                    No
5  google  computer programmer    masters                   Yes
---------------------------------------------------------
Grouped Attribute Values
   company              job     degree salary_more_then_100k
0  google  sales executive  bachelors                   No
1  google  sales executive    masters                   No
---------------------------------------------------------
---------------------------------------------------------

Total no of instances/records associated with 'business manager' is  2

Target attribute class count(Yes/No)= {'Yes': 2}

Classes Yes Yes

Probabilities of Class 'p'='Yes'  1.0
Probabilities of Class 'n'='Yes'   1.0
---------------------------------------------------------

Total no of instances/records associated with 'computer programmer' is  2

Target attribute class count(Yes/No)= {'No': 1, 'Yes': 1}

Classes Yes No

Probabilities of Class 'p'='Yes'  0.5
Probabilities of Class 'n'='No'   0.5
---------------------------------------------------------

Total no of instances/records associated with 'sales executive' is  2
```

```
Target attribute class count(Yes/No)= {'No': 2}

Classes No No

Probabilities of Class 'p'='No'  1.0
Probabilities of Class 'n'='No'   1.0
-----------------------------------------------------------

Total no of instances/records associated with 'S-google' is  6

Target attribute class count(Yes/No)= {'No': 3, 'Yes': 3}

Classes Yes No

Probabilities of Class 'p'='Yes'  0.5
Probabilities of Class 'n'='No'   0.5

Information gain of ‘‘ job ’’ is  0.6666666666666667
===========================================================


----- Information Gain Calculation of degree -----
Grouped Attribute Values
   company                  job     degree salary_more_then_100k
0  google      sales executive  bachelors                    No
2  google     business manager  bachelors                   Yes
4  google  computer programmer  bachelors                    No
-----------------------------------------------------------
Grouped Attribute Values
   company                  job   degree salary_more_then_100k
1  google      sales executive  masters                     No
3  google     business manager  masters                    Yes
5  google  computer programmer  masters                    Yes
-----------------------------------------------------------
-----------------------------------------------------------

Total no of instances/records associated with 'bachelors' is  3

Target attribute class count(Yes/No)= {'No': 2, 'Yes': 1}

Classes Yes No

Probabilities of Class 'p'='Yes'  0.6666666666666666
Probabilities of Class 'n'='No'   0.3333333333333333
-----------------------------------------------------------

Total no of instances/records associated with 'masters' is  3
```

```
Target attribute class count(Yes/No)= {'No': 1, 'Yes': 2}

Classes Yes No

Probabilities of Class 'p'='Yes'  0.6666666666666666
Probabilities of Class 'n'='No'   0.3333333333333333
-----------------------------------------------------------

Total no of instances/records associated with 'S-google' is  6

Target attribute class count(Yes/No)= {'No': 3, 'Yes': 3}

Classes Yes No

Probabilities of Class 'p'='Yes'  0.5
Probabilities of Class 'n'='No'   0.5

Information gain of '' degree '' is  0.08170416594551044
===========================================================

List of Gain for arrtibutes: ['job', 'degree']
are: [0.6666666666666667, 0.08170416594551044] respectively.


----- Information Gain Calculation of degree -----
Grouped Attribute Values
   company                  job     degree salary_more_then_100k
4  google  computer programmer  bachelors                    No
-----------------------------------------------------------
Grouped Attribute Values
   company                  job   degree salary_more_then_100k
5  google  computer programmer  masters                   Yes
-----------------------------------------------------------
-----------------------------------------------------------

Total no of instances/records associated with 'bachelors' is  1

Target attribute class count(Yes/No)= {'No': 1}

Classes No No

Probabilities of Class 'p'='No'  1.0
Probabilities of Class 'n'='No'   1.0
-----------------------------------------------------------

Total no of instances/records associated with 'masters' is  1
```

```
Target attribute class count(Yes/No)= {'Yes': 1}

Classes Yes Yes

Probabilities of Class 'p'='Yes'  1.0
Probabilities of Class 'n'='Yes'   1.0
-----------------------------------------------------------

Total no of instances/records associated with 'S-computer programmer' is  2

Target attribute class count(Yes/No)= {'No': 1, 'Yes': 1}

Classes Yes No

Probabilities of Class 'p'='Yes'  0.5
Probabilities of Class 'n'='No'   0.5

Information gain of ‘‘ degree ’’ is  1.0
===========================================================

List of Gain for arrtibutes: ['degree']
are: [1.0] respectively.

The Resultant Decision Tree is:

{'company': {'abc pharma': {'job': {'business manager': {'degree': {'bachelors':
'No',
                                                                   'masters':
'Yes'}},
                                    'computer programmer': 'No',
                                    'sales executive': 'No'}},
             'facebook': 'Yes',
             'google': {'job': {'business manager': 'Yes',
                                'computer programmer': {'degree': {'bachelors':
'No',
                                                                   'masters':
'Yes'}},
                                'sales executive': 'No'}}}}

Best Attribute  company
Tree Keys       dict_keys(['abc pharma', 'facebook', 'google'])
```

[19]:
```
ree=id3(df,t,attribute_names);
```

```
Total no of instances/records associated with 'bachelors' is  1

Target attribute class count(Yes/No)= {'No': 1}

Classes No No

Probabilities of Class 'p'='No'  1.0
Probabilities of Class 'n'='No'   1.0
-----------------------------------------------------------

Total no of instances/records associated with 'masters' is  1

Target attribute class count(Yes/No)= {'Yes': 1}

Classes Yes Yes

Probabilities of Class 'p'='Yes'  1.0
Probabilities of Class 'n'='Yes'   1.0
-----------------------------------------------------------

Total no of instances/records associated with 'S-computer programmer' is  2

Target attribute class count(Yes/No)= {'No': 1, 'Yes': 1}

Classes Yes No

Probabilities of Class 'p'='Yes'  0.5
Probabilities of Class 'n'='No'   0.5

Information gain of ‘‘degree’’ is      1.0
============================================================

List of Gain for arrtibutes: ['degree']

are: [1.0] respectively.
```

[20]:
```
import pydot

def draw(parent_name, child_name):
    edge = pydot.Edge(parent_name, child_name)
    graph.add_edge(edge)

def visit(node, parent=None):
    for k,v in node.items():
        if isinstance(v, dict):
            # We start with the root node whose parent is None
            # we don't want to graph the None node
            if parent:
```

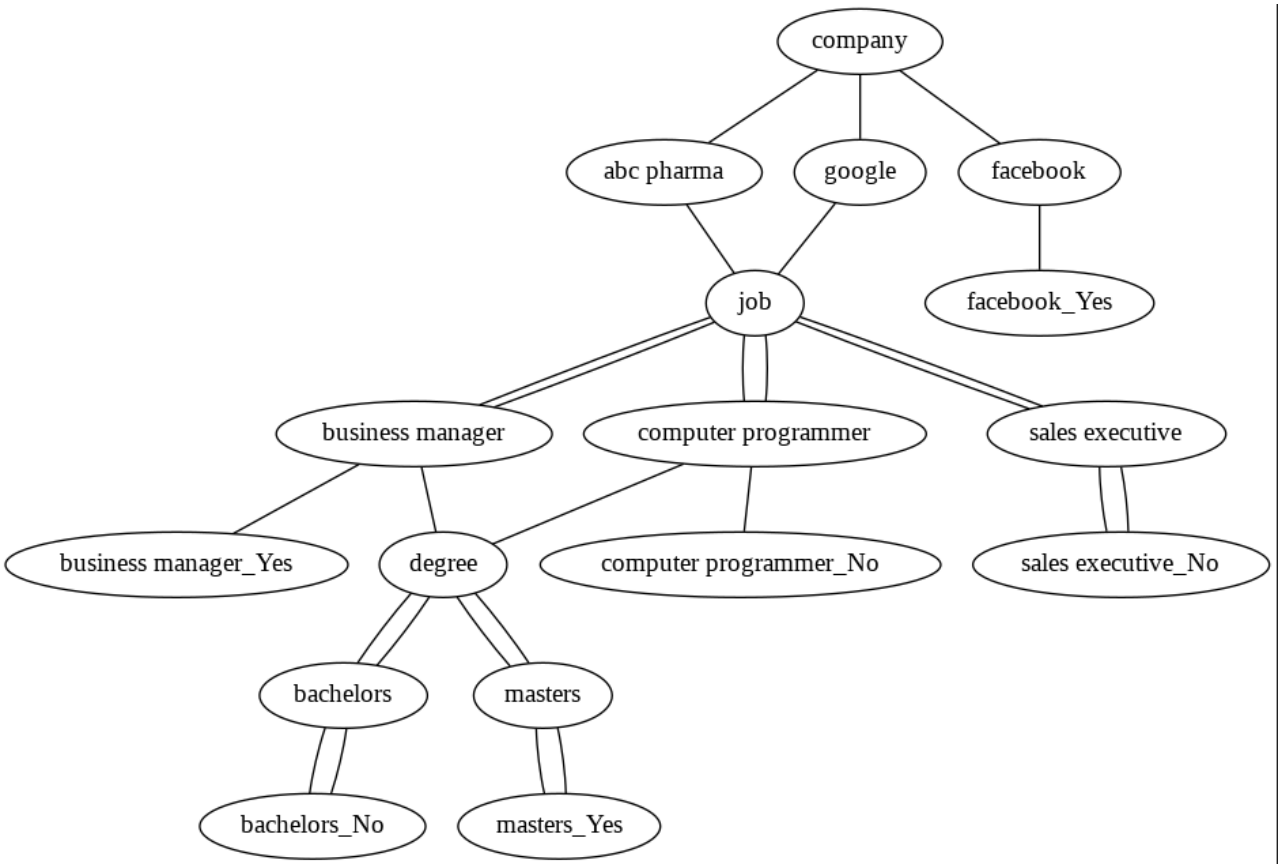
```
                draw(parent, k)
            visit(v, k)
        else:
            draw(parent, k)
            # drawing the label using a distinct name
            draw(k, k+'_'+v)

graph = pydot.Dot(graph_type='graph')
visit(ree)
graph.write_png('example1_graph.png')
```

[21]:
```
graph
```

[21]:
```
<pydot.Dot at 0x7f71b55fa2d0>
```

Output :
Root Node : company
Decision Nodes: abc pharma, google, facebook



Conclusion:

- We learned to calculate the entropy of the dataset and information gain of each attribute to decide the root node and subsequently the leaf nodes.
- A person will have salary more than 100k if :
    1. The person works at Facebook.
    2. If he works as a Business Manager at abc pharma or google.
    3. If he works as a Business Manager, Computer programmer at abc pharma, google and has a masters.

- A root node is at the beginning of a tree. It represents entire population being analyzed. From the root node, the population is divided according to various features, and those sub-groups are split in turn at each decision node under the root node.