

TITLE : 2. Candidate Elimination

Expt. No. : 2

Algorithm

Date : 21.09.2020

Harsh R Jain

(18417403)

Program:

```
import numpy as np.
```

```
import pandas as pd
```

```
data = pd.DataFrame(data=pd.read_csv('training example.csv'))
```

```
concepts = np.array(data.iloc[0:-1])
```

```
target = np.array(data.iloc[:, -1])
```

```
def learn (concepts, target):
```

```
    specific-h = concepts[0].copy()
```

```
    print('Initialisation of specific-h & general-h')
```

```
    print(specific-h)
```

```
    general-h = ["?"] * for i in range (len(specific-h)) for i in  
range (len(specific-h))
```

```
    print(general-h)
```

```
    for i, h in enumerate (concepts):
```

```
        if target[i] == 'yes':
```

```
            for x in range (len(specific-h)):
```

```
                if h[x] != specific-h[x]:
```

```
                    specific-h[x] = '?'
```

```
                    general-h[h[x]] = '?'
```

Marks :

Staff :

if target[1] == 'no':

for x in range(len(specific-h)):

if h[x] != specific-h[x]:

general-h[x] = specific-h[x]

else

general-h[x] = '?'

print('Steps = ', i+1)

print('specific-h: ', i+1)

print(specific-h)

print('general-h: ', i+1)

print(general-h)

indices = [i for i, val in enumerate(general-h) if val == '?']
- ['1', '2', '3']

prior indices

general-h.remove(['2', '?', '2', '?', '?', '?'])

return specific-h, general-h. # function ends here.

- if final, g-final: learn(concepts, target)
- print("final specific-h: ", s-final)
- print("final general-h: ", g-final)

TITLE : _____

Expt. No. : _____

Date : _____

Output:

Harsh R Jain

18417es031

Steps of candidate Elimination: 4

specific-h = 4

['Sunny', 'warm', '?', 'Strong', '?', '?']

general-h = 4

['Sunny', '?', '?', '?', '?', '?'], ['?', 'warm', '?', '?', '?', '?'],

['?', '?', '?', '?', '?', '?'], ['?', '?', '?', '?', '?', '?'],

['?', '?', '?', '?', '?', '?'], ['?', '?', '?', '?', '?', '?']

Final specific-h :

['Sunny', 'warm', '?', 'Strong', '?', '?']

Final general-h :

['Sunny', '?', '?', '?', '?', '?'], ['?', 'warm', '?', '?', '?', '?']

Marks : _____

Staff : _____

Candidate Elimination Algorithm

By Harsh R - 1BG17CS031

1. Program:

```
import numpy as np

import pandas as pd

data = pd.DataFrame(data=pd.read_csv('trainingexamples.csv'))

concepts = np.array(data.iloc[:,0:-1])

target = np.array(data.iloc[:, -1])

def learn(concepts, target):

    specific_h = concepts[0].copy()

    print("initialization of specific_h and general_h")

    print(specific_h)

    general_h = [["?" for i in range(len(specific_h))] for i in
range(len(specific_h))]

    print(general_h)

    for i, h in enumerate(concepts):

        if target[i] == "Yes":

            for x in range(len(specific_h)):

                if h[x] != specific_h[x]:

                    specific_h[x] = '?'

                    general_h[x][x] = '?'

        if target[i] == "No":

            for x in range(len(specific_h)):

                if h[x] != specific_h[x]:
```

```

        general_h[x][x] = specific_h[x]

    else:

        general_h[x][x] = '?'

    print(" steps of Candidate Elimination Algorithm",i+1)

    print("Specific_h ",i+1,"\n ")

    print(specific_h)

    print("general_h ", i+1, "\n ")

    print(general_h)


    indices = [i for i, val in enumerate(general_h) if val == ['?', '?', '?',
'?', '?', '?']]

    for i in indices:

        general_h.remove(['?', '?', '?', '?', '?', '?'])


    return specific_h, general_h

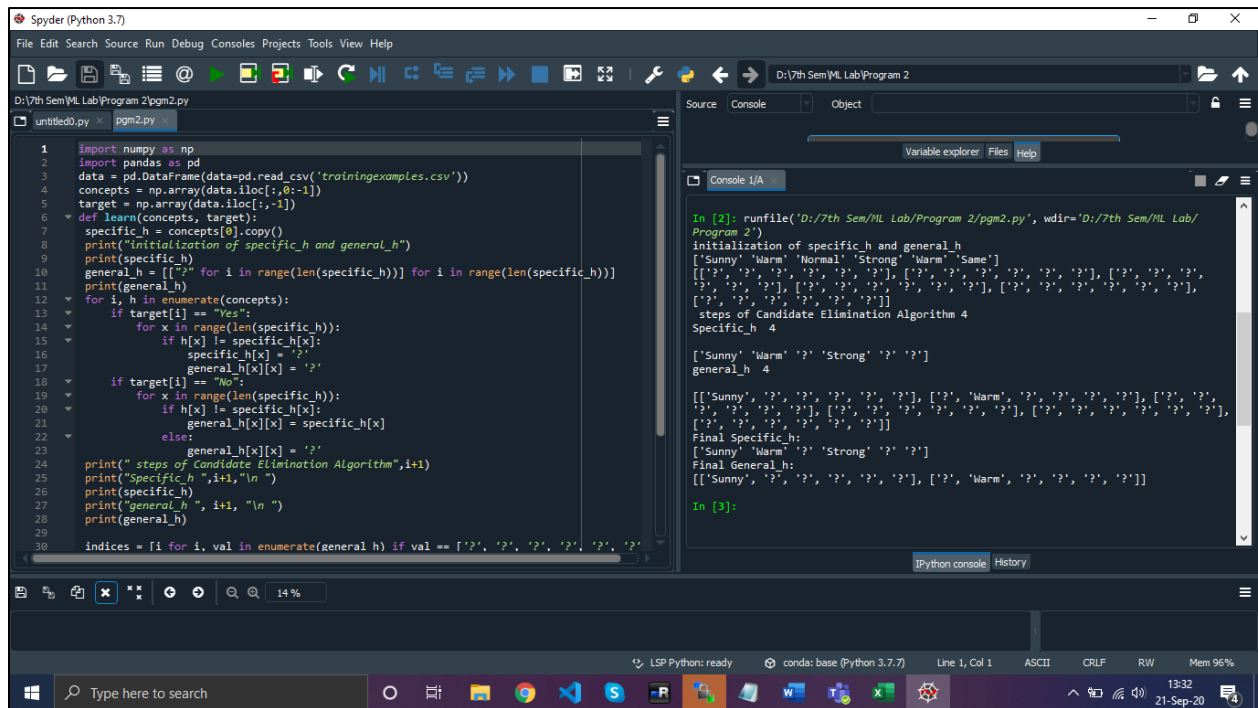

s_final, g_final = learn(concepts, target)

print("Final Specific_h:", s_final, sep="\n")

print("Final General_h:", g_final, sep="\n")

```

2. Output



The screenshot displays the Spyder Python IDE interface. The main editor window shows a Python script named `pgm2.py` with the following code:

```
1 import numpy as np
2 import pandas as pd
3 data = pd.DataFrame(data=pd.read_csv('trainingexamples.csv'))
4 concepts = np.array(data.iloc[:,0:-1])
5 target = np.array(data.iloc[:, -1])
6 def learn(concepts, target):
7     specific_h = concepts[0].copy()
8     print("Initialization of specific_h and general_h")
9     print(specific_h)
10    general_h = [['?' for i in range(len(specific_h))] for i in range(len(specific_h))]
11    print(general_h)
12    for i, h in enumerate(concepts):
13        if target[i] == "Yes":
14            for x in range(len(specific_h)):
15                if h[x] != specific_h[x]:
16                    specific_h[x] = '?'
17                    general_h[x][x] = '?'
18        if target[i] == "No":
19            for x in range(len(specific_h)):
20                if h[x] != specific_h[x]:
21                    general_h[x][x] = specific_h[x]
22                else:
23                    general_h[x][x] = '?'
24    print("Steps of Candidate Elimination Algorithm", i+1)
25    print("Specific_h ", i+1, "\n")
26    print(specific_h)
27    print("General_h ", i+1, "\n")
28    print(general_h)
29
30    indices = fi for i, val in enumerate(general_h) if val == ['?', '?', '?', '?', '?', '?']
```

The console window on the right shows the output of the script, including the initialization of `specific_h` and `general_h`, and the steps of the Candidate Elimination Algorithm. The output is as follows:

```
In [2]: runfile('D:/7th Sem/ML Lab/Program 2/pgm2.py', wdir='D:/7th Sem/ML Lab/
Program 2')
Initialization of specific_h and general_h
[['Sunny' 'Warm' 'Normal' 'Strong' 'Warm' 'Same']
[['?', '?', '?', '?', '?', '?'], ['?', '?', '?', '?', '?', '?'], ['?', '?', '?',
 '?', '?', '?'], ['?', '?', '?', '?', '?', '?'], ['?', '?', '?', '?', '?', '?'],
 ['?', '?', '?', '?', '?', '?']]
Steps of Candidate Elimination Algorithm 4
Specific_h 4
[['Sunny' 'Warm' '?' 'Strong' '?' '?']
General_h 4
[['Sunny', '?', '?', '?', '?', '?'], ['?', 'Warm', '?', '?', '?', '?'], ['?', '?',
 '?', '?', '?', '?'], ['?', '?', '?', '?', '?', '?'], ['?', '?', '?', '?', '?', '?'],
 ['?', '?', '?', '?', '?', '?']]
Final Specific_h:
[['Sunny' 'Warm' '?' 'Strong' '?' '?']
Final General_h:
[['Sunny', '?', '?', '?', '?', '?'], ['?', 'Warm', '?', '?', '?', '?']]

In [3]:
```

The bottom status bar indicates the current file is `LSP Python: ready`, the environment is `conda: base (Python 3.7.7)`, and the cursor is at `Line 1, Col 1`. The system clock shows `13:32 21-Sep-20`.