

## Exp. No.: 2

**For a given set of training data examples stored in a .CSV file, implement and demonstrate the Candidate-Elimination algorithm to output a description of the set of all hypotheses consistent with the training examples.**

### **Candidate-Elimination Algorithm:**

1. Load data set
2.  $G \leftarrow$  maximally general hypotheses in  $H$
3.  $S \leftarrow$  maximally specific hypotheses in  $H$
4. For each training example

$d = \langle x, c(x) \rangle$  Case 1 : If  $d$  is a

positive example

Remove from  $G$  any hypothesis that is  
inconsistent with  $d$  For each hypothesis  $s$  in  $S$   
that is not consistent with  $d$

- Remove  $s$  from  $S$ .
- Add to  $S$  all minimal generalizations  $h$  of  $s$  such that
- $h$  consistent with  $d$
- Some member of  $G$  is more general than  $h$
- Remove from  $S$  any hypothesis that is more general than another hypothesis in  $S$

Case 2: If  $d$  is a negative example

Remove from  $S$  any hypothesis that is  
inconsistent with  $d$  For each hypothesis  $g$  in  $G$   
that is not consistent with  $d$

- Remove  $g$  from  $G$ .
- Add to  $G$  all minimal specializations  $h$  of  $g$  such that
  - $h$  consistent with  $d$
  - Some member of  $S$  is more specific than  $h$
- Remove from  $G$  any hypothesis that is less general than another hypothesis in  $G$

### Source Code:

```
import numpy as np
import pandas as pd
data = pd.DataFrame(data=pd.read_csv('finds1.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")

```

## Output

### initialization of specific\_h and general\_h

['Cloudy' 'Cold' 'High' 'Strong' 'Warm' 'Change']

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

steps of Candidate Elimination Algorithm 8

Specific\_h 8

['?' '?' '?' 'Strong' '?' '?']

general\_h 8

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

Final Specific\_h:

['?' '?' '?' 'Strong' '?' '?']

Final General\_h:

```
['?', '?', '?', 'Strong', '?', '?']
```