

ASSIGNMENT: 01

MACHINE LEARNING – CONCEPT LEARNING

Fahad Ajmal

FA21-BSE-024

3rd March, 2024

QUESTION # 01

Applying Candidate Elimination Algorithm:

Initialize G0 and S0

S0= <NULL,NULL,NULL,NULL,NULL>

G0= <?,?,?,?,?>

Consider first training instance

X1= <Japan, Honda, Blue, 1980, Economy> (+ve)

S1= <Japan, Honda, Blue, 1980, Economy>

G1= <?,?,?,?,?>

Consider second training instance

X2= <Japan, Toyota, Green, 1970, Sports> (-ve)

S2= <Japan, Honda, Blue, 1980, Economy>

G2= <?,Honda,?,?,?> <?,?,Blue,?,?> <?,?,?,1980,?> <?,?,?,?,Economy>

Consider third training instance

X3= <Japan, Toyota, Blue, 1990, Economy> (+ve)

S3= <Japan, ?, Blue, ?, Economy>

G3= <?,?,Blue,?,?> <?,?,?,?,Economy>

Consider fourth training instance

X4= <USA, Chrysler, Red, 1980, Economy> (-ve)

S4= <Japan, ?, Blue, ?, Economy>

G4= <?,?,Blue,?,?> <Japan,?,?,?,Economy>

Consider fifth training instance

X5= <Japan, Honda, White, 1980, Economy> (+ve)

S5= <Japan,?,?,?, Economy>

G5= <Japan,?,?,?,Economy>

QUESTION # 02

Applying Find S Algorithm:

EYES	NOSE	HEAD	FACE COLOR	HAIR	EXPRESSION
round	triangle	round	purple	yes	Happy (+ve)
square	square	square	green	yes	Sad (-ve)
square	triangle	round	yellow	yes	Happy (+ve)
round	triangle	round	green	no	Sad (-ve)
square	square	round	yellow	yes	Happy (+ve)

HO: (0, 0, 0, 0, 0)

Ex1: (Round, Triangle, Round, Purple, Yes) (+ve)

H1: (Round, Triangle, Round, Purple, Yes)

Ex2: (Square, Square, Square, Green, Yes) (-ve)

H2: (Round, Triangle, Round, Purple, Yes)

Ex3: (Square, Triangle, Round, Yellow, Yes) (+ve)

H3: (?, Triangle, Round, ?, Yes)

Ex4: (Round, Triangle, Round, Green, No) (-ve)

H4: (?, Triangle, Round, ?, Yes)

Ex5: (Square, Square, Round, Yellow, Yes) (+ve)

H5: (?, ?, Round, ?, Yes)

QUESTION # 03

Python Program:

```
import numpy as np
import pandas as pd

training_examples = [

    {"size": "big", "color": "red", "shape": "circle", "output": "no"},

    {"size": "small", "color": "red", "shape": "triangle", "output": "no"},

    {"size": "small", "color": "red", "shape": "circle", "output": "yes"},

    {"size": "big", "color": "blue", "shape": "circle", "output": "no"},

    {"size": "small", "color": "blue", "shape": "circle", "output": "yes"}

]

data=pd.DataFrame(training_examples)
print(data)

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

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

def learn(concepts, target):

    specific_h = concepts[0].copy()
```

```
print("\nInitialization 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)):
```

```
            # For negative hypothesis change values only in G
```

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

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

```
else:
```

```
    general_h[x][x] = '?'
```

```
print("\nSteps of Candidate Elimination Algorithm",i+1)
```

```
print(specific_h)
```

```
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("\nFinal Specific_h:", s_final, sep="\n")
```

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