**CANDIDATE ELIMINATION ALGORITHM**

**CODE:**

import numpy as np

import pandas as pd

data = pd.DataFrame(data=pd.read\_csv('enjoysport.csv'))

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("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] ='?'

print(specific\_h)

print(specific\_h)

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)

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:**

= RESTART: C:/Users/prith/Desktop/MACHINE LEARNING/CANDIDATE ELIMINATION ALG.py

[['sunny' 'warm' 'normal' 'strong' 'warm' 'same']

['sunny' 'warm' 'high' 'strong' 'warm' 'same']

['rainy' 'cold' 'high' 'strong' 'warm' 'change']

['sunny' 'warm' 'high' 'strong' 'cool' 'change']]

['yes' 'yes' 'no' 'yes']

initialization of specific\_h and general\_h

['sunny' 'warm' 'normal' 'strong' 'warm' 'same']

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

['sunny' 'warm' 'normal' 'strong' 'warm' 'same']

['sunny' 'warm' 'normal' 'strong' 'warm' 'same']

['sunny' 'warm' 'normal' 'strong' 'warm' 'same']

['sunny' 'warm' 'normal' 'strong' 'warm' 'same']

['sunny' 'warm' 'normal' 'strong' 'warm' 'same']

['sunny' 'warm' 'normal' 'strong' 'warm' 'same']

['sunny' 'warm' 'normal' 'strong' 'warm' 'same']

steps of Candidate Elimination Algorithm 1

['sunny' 'warm' 'normal' 'strong' 'warm' 'same']

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

['sunny' 'warm' 'normal' 'strong' 'warm' 'same']

['sunny' 'warm' 'normal' 'strong' 'warm' 'same']

['sunny' 'warm' '?' 'strong' 'warm' 'same']

['sunny' 'warm' '?' 'strong' 'warm' 'same']

['sunny' 'warm' '?' 'strong' 'warm' 'same']

['sunny' 'warm' '?' 'strong' 'warm' 'same']

['sunny' 'warm' '?' 'strong' 'warm' 'same']

steps of Candidate Elimination Algorithm 2

['sunny' 'warm' '?' 'strong' 'warm' 'same']

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

['sunny' 'warm' '?' 'strong' 'warm' 'same']

steps of Candidate Elimination Algorithm 3

['sunny' 'warm' '?' 'strong' 'warm' 'same']

[['sunny', '?', '?', '?', '?', '?'], ['?', 'warm', '?', '?', '?', '?'], ['?', '?', '?', '?', '?', '?'], ['?', '?', '?', '?', '?', '?'], ['?', '?', '?', '?', '?', '?'], ['?', '?', '?', '?', '?', 'same']]

['sunny' 'warm' '?' 'strong' 'warm' 'same']

['sunny' 'warm' '?' 'strong' 'warm' 'same']

['sunny' 'warm' '?' 'strong' 'warm' 'same']

['sunny' 'warm' '?' 'strong' 'warm' 'same']

['sunny' 'warm' '?' 'strong' '?' 'same']

['sunny' 'warm' '?' 'strong' '?' '?']

['sunny' 'warm' '?' 'strong' '?' '?']

steps of Candidate Elimination Algorithm 4

['sunny' 'warm' '?' 'strong' '?' '?']

[['sunny', '?', '?', '?', '?', '?'], ['?', 'warm', '?', '?', '?', '?'], ['?', '?', '?', '?', '?', '?'], ['?', '?', '?', '?', '?', '?'], ['?', '?', '?', '?', '?', '?'], ['?', '?', '?', '?', '?', '?']]

Final Specific\_h:

['sunny' 'warm' '?' 'strong' '?' '?']

Final General\_h:

[['sunny', '?', '?', '?', '?', '?'], ['?', 'warm', '?', '?', '?', '?']]

>>>

