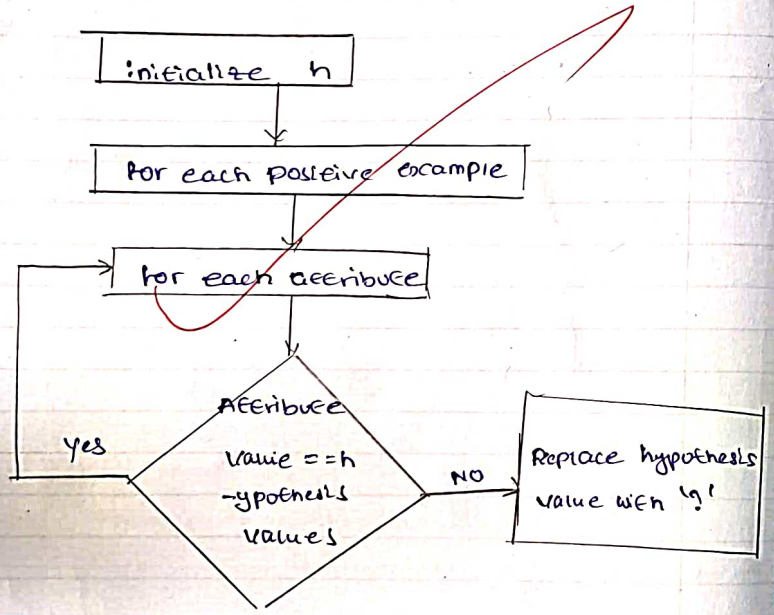
Example	Sky	AirTemp	Humidity	Wind	Water	Forecasting	Enjoy sport
1	Sunny	Warm	Normal	Strong	Warm	Same	yes
2	Sunny	warm	high	Strong	Warm	same	yes
3	sunny	Cold	high	Strong	warm	change	NO
4	Sunny	warm	high	Strong	warm	change	yes

Flowchart:-



1. Implement and demonstrate the FIND-S algorithm for finding the most specific hypothesis based on a given set of training data samples. Read the training data from a .csv file.

Aim:

Implement and demonstrate the FIND-S algorithm for finding the most specific hypothesis based on a given set of training data samples. Read the training data from a .csv file

Description:-

Consider the example taste of learning the target concept

Days on which my friend enjoys his favorite water sport

FIND-S Algorithm:-

1) Initialize h to the most specific hypothesis in H.
2) for each positive training instance x
   for each attribute constraint ai in h
   If the constraint ai is satisfied by x
   then do nothing
   else replace ai in h by the next more general constraint that is satisfied by x
3) Output hypothesis h

Conceptual learning is of two types.

1. Specific hypothesis
2. Generic hypothesis

Days on which my friend Bress.

Step 1:- Specific hypothesis which doesn't contain any specific data.

< ϕ, ϕ, ϕ, ϕ, ϕ, ϕ >

Step 2:- The first record

< Sunny, warm, normal, strong, warm, same >

Step 3:- The second record

< Sunny, warm, highest, strong, warm, same >

< Sunny, warm, ?, strong, warm, same >

Step 4:-

we will not take 3rd sample because FIND-S approach doesn't contain negative

Step 5:- Choose 4th record

< Sunny, warm, high, strong, cool, change >

specific hypothesis

Output:-

	Time	Weather	Temperature	Company	Humidity	Wind	Goes
0	Morning	Sunny	Warm	Yes	Mild	Strong	Yes
1	Evening	Rainy	Cold	no	Mild	Normal	NO
2	Morning	Sunny	moderate	yes	normal	Normal	yes
3	Evening	Sunny	cold	no	High	Strong	yes

Output:-

The attributes are:

[['morning' 'Sunny' 'warm' 'yes' 'mild' 'strong'
'Evening' 'Rainy' 'cold' 'NO' 'mild' 'Normal'
'morning' 'Sunny' 'moderate' 'yes' 'Normal' 'Normal'
'Evening' 'Sunny' 'cold' 'yes' 'High' 'strong']]

<sunny, warm, ?, strong, ?, ?>

we consider i.e as final result since, there is no more records.

program

```
import pandas as pd
data = pd.read_csv('data.csv')
data.head()
```

```
import numpy as np
concepts = np.array(data)[:,:-1]
print("The attributes are: \n", concepts)
```

output-

The target is : ['yes' 'no' 'yes' 'yes']

output

['morning' 'sunny' 'warm' 'yes' 'mild' 'strong']

['morning' 'sunny' '?' 'yes' 'mild' 'strong']

['morning' 'sunny' '?' 'yes' '?' 'strong']

['morning' 'sunny' '?' 'yes' '?' '?']

['?' 'sunny' 'yes' '?' '?']

['?' 'sunny' 'yes' '?' '?']

['?' 'sunny' 'yes' '?' '?']

['?' 'sunny' 'yes' '?' '?']

The final hypothesis is: ['?' 'sunny' '?' 'yes' '?' '?']

18/9/24

BVC
GROUP OF INSTITUTIONS
ODALAREVU • AMALAPURAM • RAJAHMUNDRY


```
target = np.array(data)[:,-1]
print("The target is:", target)


def train(c,t):
    for i, val in enumerate(t):
        if val == "yes":
            Specific_hypothesis = c[i].copy()
            break
    for i, val in enumerate(c):
        if t[i] == "yes":
            for x in range(len(Specific_hypothesis)):
                if val[x] != specific_hypothesis[x]:
                    Specific_hypothesis[x] = '?'
                    print(Specific_hypothesis)
                else:
                    pass
    return Specific_hypothesis

print("The final hypothesis is:", train(concept,train))
```

Aim:- 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 hypothesis consistent with the training examples.

Description:-

Candidate Elimination:- It is a supervised learning method used for concept learning. It maintains two sets of hypothesis

S (Specific set):- The most specific hypothesis that fits all positive examples.

G (General set):- The most general hypothesis that is consistent with all examples.

Algorithm

Steps:

1. Load Data set
2. Initialize General Hypothesis and Specific hypothesis
3. For each training example
4. if example is positive example
   if attribute value == hypothesis-value:
   Do nothing
   else: replace attribute value with '?' (Basically generalizing i.e)
5. if a example is negative example make generalize hypothesis make specific

Ex	Sky	AirTemp	Humidity	Wind	Water	Forecast	EnjoySport
0	Sunny	warm	Normal	Strong	warm	same	yes
1	Sunny	warm	High	Strong	warm	same	yes
2	Rainy	cold	High	Strong	Warm	change	NO
3	Sunny	warm	High	Strong	cool	change	yes

So $\langle \phi, \phi, \phi, \phi, \phi, \phi \rangle$

Go $\langle ?, ?, ?, ?, ?, ? \rangle$

< Sunny, warm, Normal, Strong, warm, same >

So $\langle \phi, \phi, \phi, \phi, \phi, \phi \rangle$

S1 < Sunny, warm, Normal, Strong, warm, same >

G0G1 $\langle ?, ?, ?, ?, ?, ? \rangle$

Consider the dataset and find generic and specific hypothesis using candidate elimination algorithm.

process:-

Steps

1) Read the dataset

2) At the initial stage specify most general & most specific hypothesis
They are denoted as G0 and S0.

3) In the given dataset choose the first training sample.

< Sunny, warm, Normal, Strong, warm, same >

Now Compare this sample with generic hypothesis G0 and it matches exactly

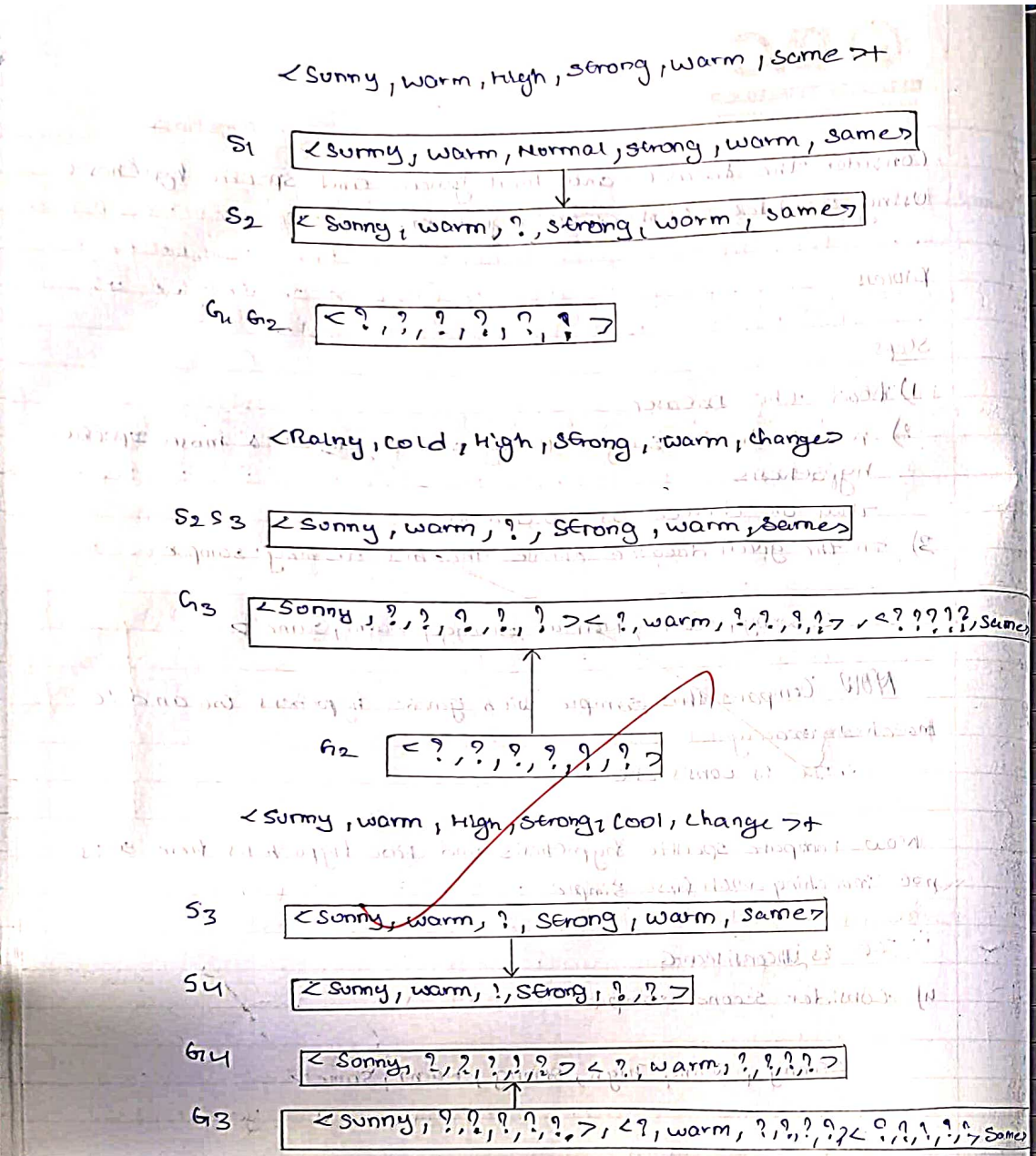
∴ It is consistent

Now compare specific hypothesis and first hypothesis here S0 is not matching with first sample.

∴ It is inconsistent

4) Consider second sample

< Sunny, warm, high, Strong, warm, same >



BVC
GROUP OF INSTITUTIONS
ODALAREVU · AMALAPURAM · RAJAHMUNDRY

here compare the G1 with second sample and it is consistent

So, compare S1 with second sample and there is mismatch.

∴ It is inconsistent

5) Consider third sample with generic hypothesis G_2 and specific hypothesis S_3

<Rainy, Cold, High, Strong, Warm, Change>

As comparing with each other there is inconsistency we try to ensure consistency.

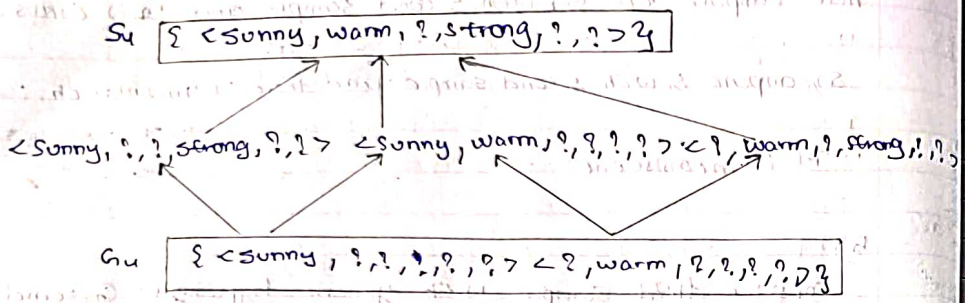
G_3 <Sunny, ?, ?, ?, ?, ?>
<?, Warm, ?, ?, ?, ?>
<?, ?, Normal, ?, ?, ?>
<?, ?, ?, Strong, ?, ?>
<?, ?, ?, ?, Cool, ?>
<?, ?, ?, ?, ?, Same>

Compare the first three sample with the above six hypothesis and remove inconsistencies.

6) Consider fourth sample

<Sunny, Warm, High, Strong, Cool, Change>

so compare fourth sample with G_3 and S_3 as performed above



Output:-

	Sky	AirTemp	Humidity	Wind	Water	Forecast	EnjoySport
0	Sunny	warm	Normal	Strong	Warm	Same	yes
1	Sunny	warm	High	Strong	Warm	Same	yes
2	Rainy	Cold	High	Strong	warm	change	No
3	Sunny	warm	High	Strong	Cold	change	yes

Initial Specific hypothesis (S): ['ø' 'ø' 'ø' 'ø' 'ø' 'ø']

Initial General hypothesis (G): [['?' '?' '?' '?' '?' '?']]

BVC
GROUP OF INSTITUTIONS
ODALAREVU · AMALAPURAM · RAJAHMUNDRY

7) The final generic and specific hypothesis are:

Gu {<Sunny, ?, ?, ?, ?, ?>, <?, warm, ?, ?, ?, ?>}

Su {<Sunny, warm, ?, strong, ?, ?>}

Program:-

```
import pandas as pd
data = pd.read_csv('sport.csv')
print("Data set:\n", data)

features = data.columns[:-1]
num_features = len(features)
S = ['ø'] * num_features
G = ['?'] * num_features
print("Initial specific hypothesis (S):", S)
print("Initial general hypothesis (G):", S)
```

Dataset:

	Sky	Airtemp	Humidity	Wind	Water	Forecase	EnjoySport
0	Sunny	Warm	Normal	Strong	Warm	Same	Yes
1	Sunny	Warm	High	Strong	Warm	Same	Yes
2	Rainy	Cold	High	Strong	Warm	Change	No
3	Sunny	Warm	High	Strong	Cold	Change	Yes

Initial Specific Hypothesis (S): ['ø', 'ø', 'ø', 'ø', 'ø', 'ø']

Initial General Hypothesis (G): [['?', '?', '?', '?', '?', '?']]

Instance 1 (Positive): ['Sunny', 'Warm', 'Normal', 'Strong', 'Warm', 'Same']

=> Yes Updated Specific Hypothesis (S): ['Sunny', 'Warm', 'Normal', 'Strong', 'Warm', 'Same']

Updated General Hypothesis (G): [['?', '?', '?', '?', '?', '?']]

Instance 2 (Positive): ['Sunny', 'Warm', 'High', 'Strong', 'Warm', 'Same']

=> Yes Updated Specific Hypothesis (S): ['Sunny', 'Warm', '?', 'Strong', 'Warm', 'Same']

Updated General Hypothesis (G): [['?', '?', '?', '?', '?', '?']]

Instance 3 (Positive): ['Sunny', 'Warm', 'High', 'Strong', 'Cold', 'Change']

=> Yes Specific Hypothesis (S): ['Sunny', 'Warm', '?', 'Strong', '?', '?']

Updated General Hypothesis (G): [['?', '?', '?', '?', '?', '?']]


```
# Display Dataset and hypothesis
print("\n Dataset:\n", data)
print("\n Initial Specific Hypothesis (S):", S)
print("Initial General Hypothesis (G):", G)



# Process positive instances
for i, row in data.iterrows():
    instance = row[:-1]
    target = row[-1]
    if target == 'yes':
        for idx, feature_val in enumerate(instance):
            if S[idx] == 'ø':
                S[idx] = feature_val
            elif S[idx] != feature_val:
                S[idx] = '?'
        G = [g for g in G if all(x == '?' or x == y for x, y in
                                 zip(g, S))]
    print(f"\n Instance {i+1} (Positive): {list(instance)}
                                 => {target} ")
print("Updated Specific Hypothesis (S):", S)
print("Updated General Hypothesis (G):", G)
```

Instance 3 (Negative): ['Rainy', 'cold', 'High', 'strong', 'warm', 'change'] => No updated specific hypothesis (S): ['Sunny', 'warm', '?', 'strong', '?', '?']

Updated General hypothesis (G): [['Sunny', '?', '?', '?', '?', '?'], ['?', 'warm', '?', '?', '?', '?']]


```
print("-" * 50)

# Process negative instances
for i, row in data.iterrows():
    instance = row[:-1]
    target = row[-1]
    if target == 'No':
        G_new = []
        for g in G:
            if all(x == '?' or x == val for x, val in zip(g, instance)):
                for idx, val in enumerate(g):
                    if val == '?':
                        for feature_val in data[features[idx]].unique():
                            if feature_val != instance[idx]:
                                new_hypothesis = g.copy()
                                new_hypothesis[idx] = feature_val
                                if all(x == '?' or x == y for x, y in zip(new_hypothesis, S)):
                                    G_new.append(new_hypothesis)
        G = G_new
print(f"\nInstance {i+1} (Negative): {dict(instance)} => {target}")
print("Updated specific hypothesis (S):", S)
```

Final Specific hypothesis (S): ['Sunny', 'warm', '?', 'strong', '?', '?']

Final General hypothesis (G): [['Sunny', '?', '?', '?', '?', '?'], ['?', 'Warm', '?', '?', '?', '?']]

19/9/24

print ("Updated General Hypothesis (G):", G)
print ("-" * 50)

Output the final hypotheses
print ("In Final Specific hypothesis (S):", S)
print ("Final General hypothesis (G):", G)