

EX.NO:1	FIND S ALGORITHM
DATE:	

AIM:

To implement a Find S Algorithm for bank marketing.

PROCEDURE:

1. Import the pandas library.
2. Load the reduced dataset (bank_reduced.csv) using read_csv().
3. Separate the dataset into:
 - o X → attribute values (all columns except the last)
 - o y → class label (deposit)
4. Initialize the hypothesis with the first positive example (deposit = yes).
5. Compare the hypothesis with all other positive examples:
 - o If an attribute value differs, replace it with "?".
6. Ignore all negative examples (deposit = no).
7. Continue until all training examples are processed.
8. Display the final hypothesis.

DATASET DESCRIPTION:

The dataset used for this experiment is bank_reduced.csv, which is a preprocessed and reduced version of the Bank Marketing dataset.

Each row represents a bank customer, and each column represents a categorical attribute related to the customer.

Attributes Used:

Attribute	Description
job	Type of job of the customer
marital	Marital status
education	Education level
housing	Housing loan (yes/no)
loan	Personal loan (yes/no)

Attribute	Description
contact	Contact communication type
month	Month of last contact
deposit	Target class – subscription to term deposit (yes/no)

- The dataset contains only categorical values.
- The class label is deposit.
- The dataset is reduced to a small number of instances for concept learning using Find-S.

PROGRAM:

```

import pandas as pd

# Load the reduced dataset

df = pd.read_csv("bank_reduced.csv")

# Separate attributes and class label

X = df.iloc[:, :-1].values # all columns except last

y = df.iloc[:, -1].values # last column: deposit

# Find-S algorithm implementation

def find_s(X, y):

    # Initialize hypothesis as the first positive example

    hypothesis = None

    for i in range(len(y)):

        if y[i] == "yes":

            hypothesis = X[i].copy()

            break

    # Loop over all examples

    for i in range(len(y)):

        if y[i] == "yes":

            for j in range(len(hypothesis)):

                if X[i][j] != hypothesis[j]:

                    hypothesis[j] = "?"

```

```

return hypothesis

# Run Find-S

final_hypothesis = find_s(X, y)

print("Final Hypothesis (Find-S):")

print(final_hypothesis)

```

RESULT/OUTPUT:



```

import pandas as pd
# Load the reduced dataset
df = pd.read_csv("bank_reduced.csv")
# Separate attributes and class label
X = df.iloc[:, :-1].values # all columns except last
y = df.iloc[:, -1].values # last column: deposit
# Find-S algorithm implementation
def find_s(X, y):
    # Initialize hypothesis as the first positive example
    hypothesis = None
    for i in range(len(y)):
        if y[i] == "yes":
            hypothesis = X[i].copy()
            break
    # Loop over all examples
    for i in range(len(y)):
        if y[i] == "yes":
            for j in range(len(hypothesis)):
                if X[i][j] != hypothesis[j]:
                    hypothesis[j] = "?"
    return hypothesis
# Run Find-S
final_hypothesis = find_s(X, y)
print("Final Hypothesis (Find-S):")
print(final_hypothesis)

...
*** Final Hypothesis (Find-S):
['?' '?' '?' '?' '?' 'unknown' 'may']

```

RESULT:

Thus the Find S Algorithm predicts matches the positive examples, with mismatched features replaced by (?) has been executed and output was verified successfully.