

```

import csv

import copy

from tkinter import Tk

from tkinter.filedialog import askopenfilename


def load_csv_file():
    """Open a dialog to select CSV file and load data"""

    Tk().withdraw() # Hide the root Tkinter window

    filename = askopenfilename(filetypes=[("CSV files", "*.csv")], title="Select the training CSV file")

    if not filename:

        print("No file selected. Exiting.")

        exit()


    with open(filename, 'r') as file:

        data = list(csv.reader(file))

    return data


def candidate_elimination(data):
    """Candidate Elimination Algorithm"""

    attributes = data[0][:-1] # attribute names (ignore last column)

    examples = data[1:]      # all rows except header


    # Initialize S and G

    S = ['∅'] * len(attributes) # most specific hypothesis

    G = [['?'] * len(attributes)] # most general hypothesis


    print("Initial Specific boundary (S):", S)

    print("Initial General boundary (G):", G)

    print("="*60)


    for example in examples:

```

```

x = example[:-1] # attribute values
label = example[-1] # target value (Yes/No)

if label.lower() == 'yes':
    # Remove inconsistent hypotheses from G
    G = [g for g in G if all(g[i] == '?' or g[i] == x[i] for i in range(len(x)))]

    # Generalize S
    for i in range(len(x)):
        if S[i] == '∅':
            S[i] = x[i]
        elif S[i] != x[i]:
            S[i] = '?'

else: # Negative example
    new_G = []
    for g in G:
        # Check if g covers negative example
        if all(g[i] == '?' or g[i] == x[i] for i in range(len(x))):
            for i in range(len(x)):
                if g[i] == '?':
                    if S[i] != '∅' and S[i] != x[i]:
                        new_hypothesis = copy.deepcopy(g)
                        new_hypothesis[i] = S[i]
                        if new_hypothesis not in new_G:
                            new_G.append(new_hypothesis)
            else:
                if g not in new_G:
                    new_G.append(g)
    G = new_G

```

```
# Display S and G after each example
```

```
print("Example:", example)
```

```
print("Specific boundary (S):", S)
```

```
print("General boundary (G):", G)
```

```
print("-"*60)
```

```
return S, G
```

```
# ----- Main Program -----
```

```
data = load_csv_file()
```

```
S_final, G_final = candidate_elimination(data)
```

```
print("\nFinal Specific Boundary (S):", S_final)
```

```
print("Final General Boundary (G):", G_final)
```

OUTPUT:

```
IDLE Shell 3.13.9
File Edit Shell Debug Options Window Help
Python 3.13.9 (tags/v3.13.9:8183fa5, Oct 14 2025, 14:09:13) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>>
===== RESTART: C:/Users/prast/OneDrive/Desktop/ML LAB PDF/EXP 2 =====
Initial Specific boundary (S): ['e', 'e', 'e', 'e', 'e']
Initial General boundary (G): [['?', '?', '?', '?', '?']]
=====
Example: ['Sunny', 'Warm', 'Normal', 'Strong', 'Warm', 'Same', 'Yes']
Specific boundary (S): ['Sunny', 'Warm', 'Normal', 'Strong', 'Warm', 'Same']
General boundary (G): [['?', '?', '?', '?', '?']]
-----
Example: ['Sunny', 'Warm', 'High', 'Strong', 'Warm', 'Same', 'Yes']
Specific boundary (S): ['Sunny', 'Warm', '?', 'Strong', 'Warm', 'Same']
General boundary (G): [['?', '?', '?', '?', '?']]
-----
Example: ['Rainy', 'Cold', 'High', 'Strong', 'Warm', 'Change', 'No']
Specific boundary (S): ['Sunny', 'Warm', '?', 'Strong', 'Warm', 'Same']
General boundary (G): [['Sunny', '?', '?', '?', '?'], ['?', 'Warm', '?', '?', '?'], ['?', '?', '?', '?', '?'], ['?', '?', '?', '?', 'Same']]
-----
Example: ['Sunny', 'Warm', 'High', 'Strong', 'Cool', 'Change', 'Yes']
Specific boundary (S): ['Sunny', 'Warm', '?', 'Strong', '?', '?']
General boundary (G): [['Sunny', '?', '?', '?', '?'], ['?', 'Warm', '?', '?', '?'], ['?', '?', '?', '?', '?']]
-----
Final Specific Boundary (S): ['Sunny', 'Warm', '?', 'Strong', '?', '?']
Final General Boundary (G): [['Sunny', '?', '?', '?', '?'], ['?', 'Warm', '?', '?', '?'], ['?', '?', '?', '?', '?']]
>>>
```