Bharatiya Vidya Bhavan's
Sardar Patel Institute of Technology
Bhavan's Campus, Munshi Nagar, Andheri (West), Mumbai-400058-India
(Autonomous College Affiliated to University of Mumbai)

BE-ETRX B **Sub- AIML Lab**
Name: Shubham Sawant **UID :2019110050**

Name of the Experiment: Candidate elimination algorithm

Outcomes:
1. Representation of version space.
2. Apply the Candidate Elimination algorithm on the given data to get version space.
3. Interpret the output of Candidate Elimination.

System Requirements: Windows with MATLAB

Data Set Link: Link

Dataset Description:

Number of Instances: 5

Number of Attributes: 7

Attribute Information:

1. Engine: Type of Engine
2. Gear: No. of Gears
3. Tyre: Type of Tyre
4. Type: Type of Car
5. Roof: Type of Roof
6. Brakes: Types of Brake
7. y: Whether to buy a car or not

Problem Statement:

Concept: Is exchangeable? YES – Positive example, NO – Negative Example.

Algorithm:
The candidate elimination algorithm incrementally builds the version space given a hypothesis space H and a set E of examples. The examples are added one by one; each example possibly shrinks the version space by removing the hypotheses that are inconsistent with the example. The candidate elimination algorithm does this by updating the general and specific boundary for each new example.
Procedure: CandidateEliminationLearner(X,Y,E,H)
Inputs
    X: set of input features, X={X1,...,Xn}
    Y: target feature
    E: set of examples from which to learn
    H: hypothesis space
Output
    general boundary G⊆H
    specific boundary S⊆H consistent with E
Local
    G: set of hypotheses in H
    S: set of hypotheses in H
Let G={true}, S={false};

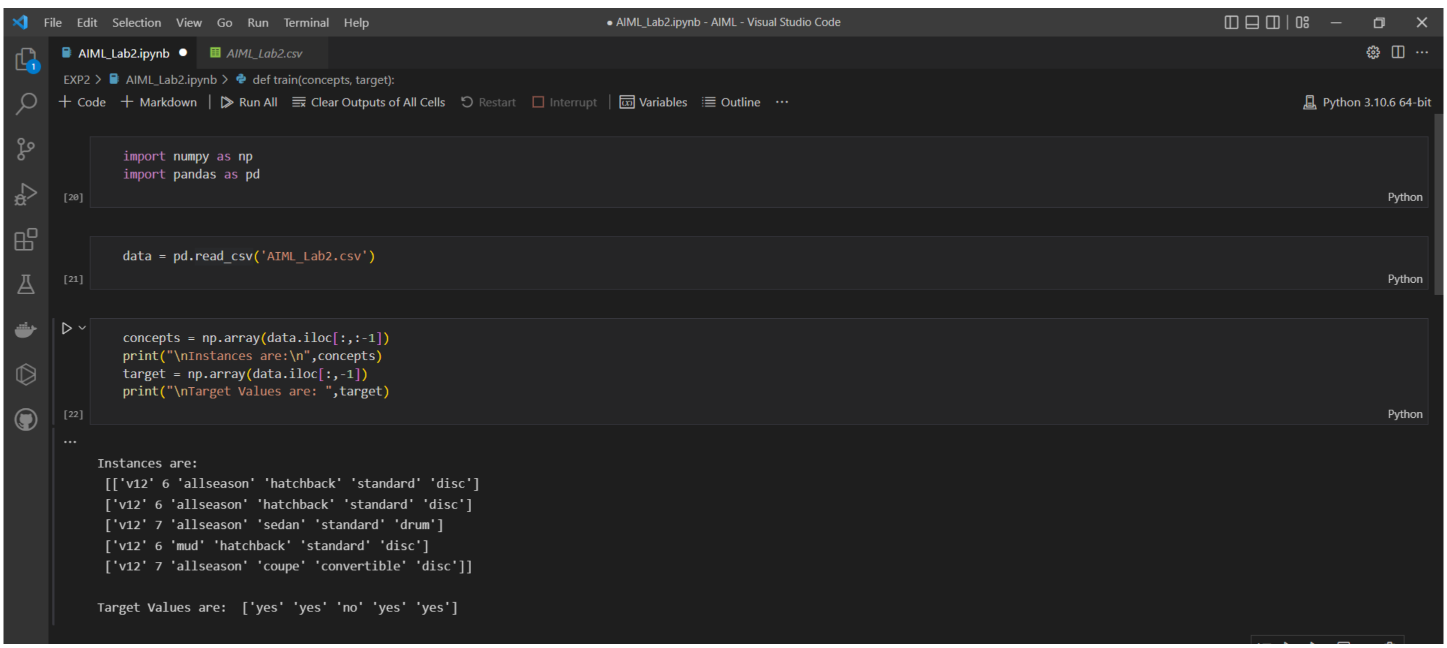
for each e∈E do
    if (e is a positive example) then
        Elements of G that classify e as negative are removed from G;
        Each element s of S that classifies e as negative is removed and replaced by the minimal generalizations of s that classify e as positive and are less general than some member of G;
        Non-maximal hypotheses are removed from S;
    else
        Elements of S that classify e as positive are removed from S;
        Each element g of G that classifies e as positive is removed and replaced by the minimal specializations of g that classifies e as negative and are more general than some members of S.
        Non-minimal hypotheses are removed from G.

Properties of Candidate Elimination :

FIND-S outputs a hypothesis from H that is consistent with the training examples; this is just one of many hypotheses from H that might fit the training data equally well. The key idea in the Candidate-Elimination algorithm is to output a description of the set of all hypotheses consistent with the training examples.

1. Candidate-Elimination algorithm computes the description of this set without explicitly enumerating all of its members.
2. This is accomplished by using the more-general-than partial ordering and maintaining a compact representation of the set of consistent hypotheses.

Code:



```
import numpy as np
import pandas as pd
```

```
data = pd.read_csv('AIML_Lab2.csv')
```

```
concepts = np.array(data.iloc[:,:-1])
print("\nInstances are:\n",concepts)
target = np.array(data.iloc[:,-1])
print("\nTarget Values are: ",target)
```

```
Instances are:
 [['v12' 6 'allseason' 'hatchback' 'standard' 'disc']
 ['v12' 6 'allseason' 'hatchback' 'standard' 'disc']
 ['v12' 7 'allseason' 'sedan' 'standard' 'drum']
 ['v12' 6 'mud' 'hatchback' 'standard' 'disc']
 ['v12' 7 'allseason' 'coupe' 'convertible' 'disc']]

Target Values are:  ['yes' 'yes' 'no' 'yes' 'yes']
```

Bharatiya Vidya Bhavan's
Sardar Patel Institute of Technology

Bhavan's Campus, Munshi Nagar, Andheri (West), Mumbai-400058-India

(Autonomous College Affiliated to University of Mumbai)

BE-ETRX B **Sub- AIML Lab**

Name: Shubham Sawant **UID :2019110050**

AIML_Lab2.ipynb - AIML - Visual Studio Code

EXP2 > AIML_Lab2.ipynb > def train(concepts, target):

```python
def train(concepts, target):

    # Initializing general and specific hypothesis
    specific_h = concepts[0].copy()
    print("\nInitialization of specific hypothesis and general hypothesis")
    print("\nSpecific Boundary: ", specific_h)
    general_h = [["?" for i in range(len(specific_h))] for i in range(len(specific_h))]
    print("\nGeneric Boundary: ",general_h)


    for i, val in enumerate(concepts):
        # print("\nInstance", i+1 , "is ", val)
        #positive example
        if target[i] == "yes":
            # print("Instance is Positive ")
            for x in range(len(specific_h)):
                if val[x]!= specific_h[x]:
                    specific_h[x] ='?'
                    general_h[x][x] ='?'
        #negative example
        if target[i] == "no":
            # print("Instance is Negative ")
            for x in range(len(specific_h)):
                if val[x]!= specific_h[x]:
                    general_h[x][x] = specific_h[x]
                else:
                    general_h[x][x] = '?'

        # print("Specific Bundary after ", i+1, "Instance is ", specific_h)
        # print("Generic Boundary after ", i+1, "Instance is ", general_h)
        # print("\n")

    indices = [i for i, val in enumerate(general_h) if val == ['?', '?', '?', '?', '?', '?']]

    for i in indices:
```

AIML_Lab2.ipynb - AIML - Visual Studio Code

EXP2 > AIML_Lab2.ipynb > def train(concepts, target):

```python
        # print("Generic Boundary after ", i+1, "Instance is ", general_h)
        # print("\n")

    indices = [i for i, val in enumerate(general_h) if val == ['?', '?', '?', '?', '?', '?']]

    for i in indices:
        general_h.remove(['?', '?', '?', '?', '?', '?'])

    return specific_h, general_h
```

[23] Python

```python
s_final, g_final = train(concepts, target)
# displaying Specific_hypothesis
print("Final Specific_h: ", s_final, sep="\n")
# displaying Generalized_Hypothesis
print("Final General_h: ", g_final, sep="\n")
```

[24] Python

```
Initialization of specific hypothesis and general hypothesis

Specific Boundary:  ['v12' 6 'allseason' 'hatchback' 'standard' 'disc']

Generic Boundary:  [['?', '?', '?', '?', '?', '?'], ['?', '?', '?', '?', '?', '?'], ['?', '?', '?', '?', '?', '?'], ['?', '?', '?', '?', '?', '?'], ['?', '?', '?', '?', '?', '?'],
['?', '?', '?', '?', '?', '?']]
Final Specific_h:
['v12' '?' '?' '?' '?' 'disc']
Final General_h:
[['?', '?', '?', '?', '?', 'disc']]
```

BE-ETRX B **Sub- AIML Lab**
Name: Shubham Sawant **UID :2019110050**

Application:

Q1) Compare between inductive and deductive learning
Ans)
Inductive learning creates a generalisation from the input data. We basically try to map the relationship between the input and output. It is a bottom-up approach and is used in various industrial applications. In this method we derive conclusions or generalize it from the given facts i.e. the data. It is similar to supervised learning. One example is Find-S algorithm where only positive samples are used to generalise to the data.

Deductive learning is similar to inductive learning. It also tries to fit in a hypothesis, but in addition it also relies on already available information to derive facts and information to give a valid conclusion. It is a top-down approach. The one major thing to note is that in deductive learning, the results are certain i.e, it is either yes or no. Whereas it's probability based on inductive learning i.e, it can range from strong to weak .

Q2) Justify candidate elimination is deductive learning
Ans)
In Candidate learning algorithm, both positive and negative samples are considered because it uses all available information and facts to provide a conclusion. It has positive samples for better generalisation and negative samples to prevent overgeneralisation.

It is a step-by-step process where each sample is compared with previous specific hypothesis if it is a positive one and with previous general if it is a negative one. Thus we get final specific and general hypothesis from the algorithm and then we can find the version space by getting all common values from the final hypothesis.

Conclusion:

1. Therefore, we can conclude that out of all the 6 attributes except the target attribute, 4 of them are generalised and we use '?' as a notation there and for other attributes the specific values must be taken for satisfying the hypothesis.
2. For the general hypothesis in case of negative samples, previous general hypothesis is compared with the current specific and common ones are rejected while different ones are listed in the general hypothesis. We do this for all the samples one by one.
3. Thus, we learned how Candidate Elimination algorithm works. We also implemented the algorithm and calculated our hypothesis.