

Aprendizado de Conceito

EXAMPLE OF A CONCEPT LEARNING TASK

Day	Sky	Airtemp	Humidity	Wind	Water	Forecast	WaterSport
1.	Sunny	Warm	Normal	Strong	Warm	Same	Yes
2.	Sunny	Warm	High	Strong	Warm	Same	Yes
3.	Rainy	Cold	High	Strong	Warm	Change	No
4.	Sunny	Warm	High	Strong	Cool	Change	Yes

Chosen Hypothesis Representation:

Conjunction of constraints on each attribute where:

- “?” means “any value is acceptable”
- “0” means “no value is acceptable”

Example of a hypothesis: <?,Cold,High,?,?,?>

(If the air temperature is cold and the humidity high then it is a good day for water sports)

David Menotti

www.inf.ufpr.br/menotti/ci171-182

Aprendizagem de Conceito

Agenda

- Introdução
- Modelando um Problema
- Busca
- Algoritmo Find-S
- Algoritmo Eliminação de Candidato
- Resumo

Introdução

- Obtenção da definição de uma categoria geral a partir de exemplos de treinamento positivos e negativos da categoria.
- Pode ser formulado como um problema de busca em um espaço de hipóteses pré-definido
- Encontrar a hipótese que melhor se ajusta aos exemplos de treinamento.

Introdução

- Aprendizagem envolve aquisição de conceitos gerais a partir de exemplos de treinamento específicos.
 - Pessoas aprendem conceitos gerais ou categorias como por exemplo, pássaro, carro, etc..
- Em aprendizagem de conceito o problema consiste em inferir automaticamente a definição geral de alguns conceitos, a partir de exemplos rotulados com **membro** ou **não membro** do conceito

Problema

- Considere o problema de aprender o conceito
 - “dias bons para praticar esporte”
- A tabela abaixo apresenta um conjunto de dias, cada um representado por um conjunto de atributos (características)

Sky	Temp	Humid	Wind	Water	Forecst	EnjoySpt
Sunny	Warm	Normal	Strong	Warm	Same	Yes
Sunny	Warm	High	Strong	Warm	Same	Yes
Rainy	Cold	High	Strong	Warm	Change	No
Sunny	Warm	High	Strong	Cool	Change	Yes

Treinamento

- No treinamento queremos encontrar a relação entre:
 - “valor dos atributos” e o valor do atributo alvo.
- Uma vez treinado, dado um dia qualquer (representado pelas características), deseja-se saber o valor do conceito alvo.

Sky	Temp	Humid	Wind	Water	Forecst	EnjoySpt
Sunny	Warm	Normal	Strong	Warm	Same	Yes
Sunny	Warm	High	Strong	Warm	Same	Yes
Rainy	Cold	High	Strong	Warm	Change	No
Sunny	Warm	High	Strong	Cool	Change	Yes

Treinamento

Tarefa da aprendizagem consiste em aprender a prever o valor de **EnjoySport** para um dia qualquer baseado somente nos valores das seis características.

- Qual representação que deve ser usada?
- Representação simples em que cada hipótese consiste em uma conjunção de restrições sobre os atributos de entrada.
- Cada hipótese é vista como um vetor de restrições especificando os valores dos seis atributos.

[Sky, Temp, Humidity, Wind, Water, Forecast]

Treinamento

- Para cada atributo, temos as seguintes opções:
 - **?**: Indica que qualquer valor é aceitável
 - **∅**: Indica que nenhum valor é aceitável
 - Valor único para o atributo, por exemplo, **Warm** para o atributo **Temp**
- Se alguma instância **x** satisfizer todas as restrições da hipótese **h**, então, **h** classifica **x** como um exemplo positivo
 - $h(x) = 1$
- Se uma instância **x** não satisfizer todas as restrições da hipótese **h**, então, **h** classifica **x** como um exemplo negativo
 - $h(x) = 0$

Exemplo

- A hipótese que dias úmidos com alta umidade são bons para a prática de esporte poderia ser assim apresentado:
 - [?, Cold, High, ?, ?, ?]
- A hipótese mais geral seria
 - [?, ?, ?, ?, ?, ?]
 - Qualquer situação é boa para pratica de esporte.
- A mais específica seria:
 - [∅, ∅, ∅, ∅, ∅, ∅]
 - Nenhum dia é bom para a prática de esporte

Exemplo

- Em resumo, qualquer tarefa de aprendizagem de conceito pode ser descrita por:
 - Um conjunto de hipóteses candidatas
 - [?, Cold, High, ?, ?, ?]
 - [Sunny, ?, High, ?, ?, ?]
 - [Sunny, ?, ?, ?, ?, ?]
 - Um conceito alvo
 - Um conjunto de exemplos de treinamento.

Aprendizagem como tarefa de Busca

- A aprendizagem de conceito pode ser vista como uma tarefa de busca em um grande espaço de hipóteses definido pela representação das mesmas.
- O objetivo desta busca é encontrar a hipótese que melhor se ajusta aos exemplos de treinamento.
- Diversos algoritmos podem ser utilizados com essa finalidade

Aprendizagem como tarefa de Busca

- Muitos algoritmos de aprendizagem de conceito organizam a busca no espaço de hipóteses baseando-se na ordenação das hipóteses
- Sentido da Busca
 - Específicas \rightarrow Gerais
- Exemplo:
 - $h_1 = [\text{Sunny}, ?, ?, \text{Strong}, ?, ?]$
 - $h_2 = [<\text{Sunny}, ?, ?, ?, ?, ?]$
 - $h_3 = [<\text{Sunny}, ?, ?, ?, \text{Cool}, ?]$
- Temos que $h_2 \geq_g h_3 \geq_g h_1$ ou $h_2 \geq_g h_1 \geq_g h_3$

Find-S

Usando o conceito de ordenação, o algoritmo começa com a hipótese **mais específica possível** e então generaliza cada vez que ela falhar em cobrir um exemplo positivo.

1. Inicializar h com a hipótese mais específica do conjunto de hipóteses H
2. Para cada exemplo positivo de treinamento x
 - a. Para cada restrição de atributo a_i em h
 - i. Se a restrição a_i não é satisfeita por x
troque a_i em h pela próxima restrição mais geral que é satisfeita por x
3. Retorna a hipótese h

Find-S

Exemplo

- Passo 1: Inicializar h com a hipótese mais específica

$$- h_0 = [\emptyset, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset]$$

Sky	Temp	Humid	Wind	Water	Forecst	EnjoySpt
Sunny	Warm	Normal	Strong	Warm	Same	Yes
Sunny	Warm	High	Strong	Warm	Same	Yes
Rainy	Cold	High	Strong	Warm	Change	No
Sunny	Warm	High	Strong	Cool	Change	Yes

h é muito específica, pois não satisfaz nenhuma instância da base de treinamento.

Find-S

Exemplo

- Passo 2(i): Trocar as restrições pela próxima restrição mais geral que se ajusta ao exemplo
 - $h_1 = [\text{Sunny}, \text{Warm}, \text{Normal}, \text{Strong}, \text{Warm}, \text{Same}]$

Sky	Temp	Humid	Wind	Water	Forecst	EnjoySpt
Sunny	Warm	Normal	Strong	Warm	Same	Yes
Sunny	Warm	High	Strong	Warm	Same	Yes
Rainy	Cold	High	Strong	Warm	Change	No
Sunny	Warm	High	Strong	Cool	Change	Yes

h ainda é bastante específica. Satisfaz apenas uma instância da base de treinamento

Find-S

Exemplo

- Passo 2(ii): Trocar as restrições pela próxima restrição mais geral que se ajusta ao exemplo
 - $h_2 = [\text{Sunny}, \text{Warm}, ?, \text{Strong}, \text{Warm}, \text{Same}]$

Sky	Temp	Humid	Wind	Water	Forecst	EnjoySpt
Sunny	Warm	Normal	Strong	Warm	Same	Yes
Sunny	Warm	High	Strong	Warm	Same	Yes
Rainy	Cold	High	Strong	Warm	Change	No
Sunny	Warm	High	Strong	Cool	Change	Yes

h agora satisfaz duas instâncias do treinamento

Find-S

Exemplo

- Passo 2(iii): O algoritmo ignora o exemplo negativo
 - $h_3 = [\text{Sunny}, \text{Warm}, ?, \text{Strong}, \text{Warm}, \text{Same}]$

Sky	Temp	Humid	Wind	Water	Forecst	EnjoySpt
Sunny	Warm	Normal	Strong	Warm	Same	Yes
Sunny	Warm	High	Strong	Warm	Same	Yes
Rainy	Cold	High	Strong	Warm	Change	No
Sunny	Warm	High	Strong	Cool	Change	Yes

Find-S

Exemplo

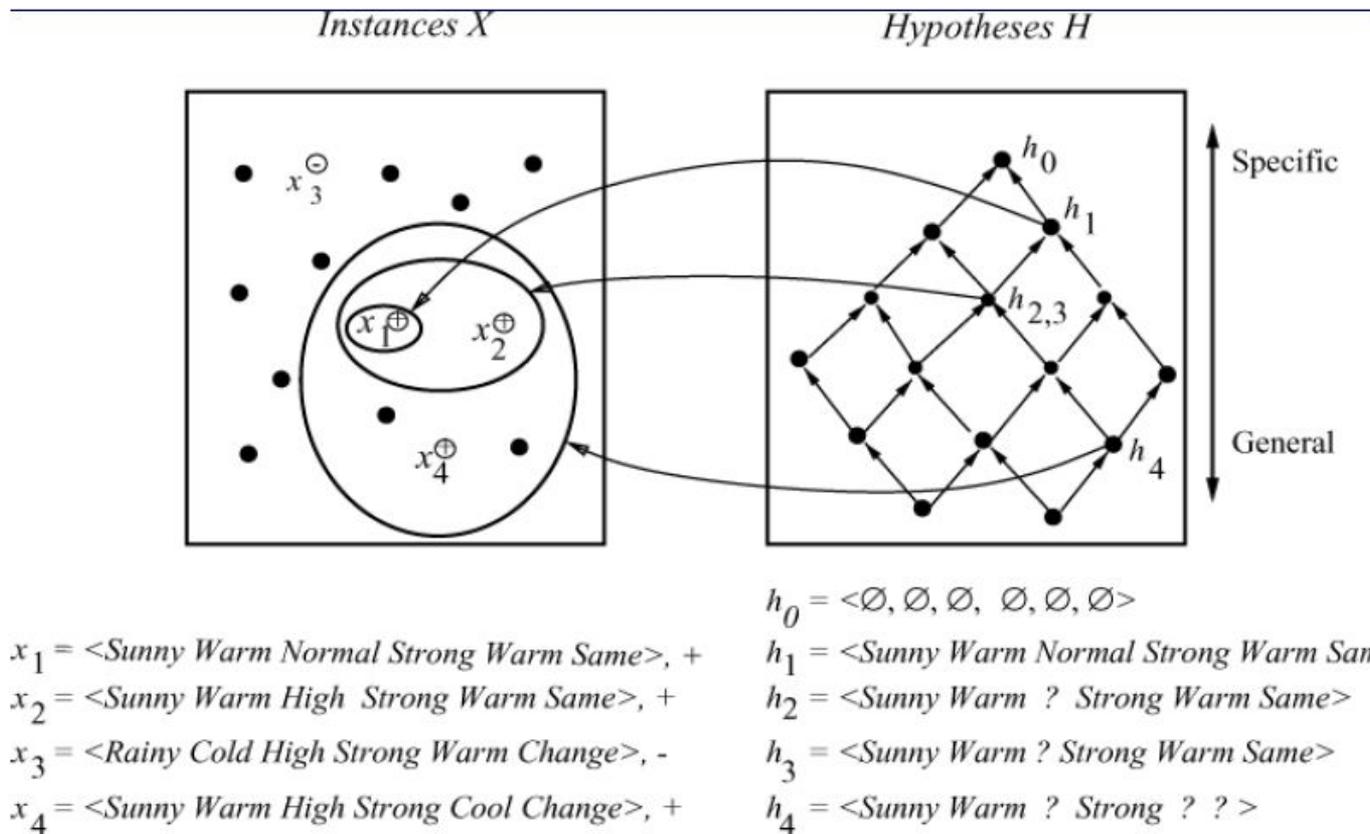
- Passo 2(iv): O quarto exemplo (3o. positivo) leva a
 - $h_4 = [\text{Sunny}, \text{Warm}, ?, \text{Strong}, ?, ?]$

Sky	Temp	Humid	Wind	Water	Forecst	EnjoySpt
Sunny	Warm	Normal	Strong	Warm	Same	Yes
Sunny	Warm	High	Strong	Warm	Same	Yes
Rainy	Cold	High	Strong	Warm	Change	No
Sunny	Warm	High	Strong	Cool	Change	Yes

- Esse algoritmo ilustra como a ordenação parcial (mais geral que) pode ser usada para organizar a busca por uma hipótese aceitável

Find-S

Busca no espaço de hipóteses



Find-S

Propriedade Básica: Para espaços de hipóteses descritos pela conjunção de atributos é garantido que **Find-S** produza a hipótese mais específica dentro de H que é consistente com os exemplos de treinamento positivos.

Entretanto:

- Não se pode dizer que o algoritmo convergiu para o conceito correto.
- Pode haver diversas outras hipóteses específicas consistentes.
- Outros algoritmos
 - Eliminação de candidatos
 - Busca exaustiva

Eliminação do Candidatos

- A idéia chave do algoritmo Eliminação de Candidatos é fornecer uma descrição do conjunto de todas as hipóteses consistentes com os exemplos de treinamento.
- Também utiliza a propriedade do ordenamento parcial **mais geral que**
- Encontra todas as hipóteses “descritíveis” que são consistentes com os exemplos de treinamento observados.
- **Definição:** Uma hipótese h é consistente com os exemplos de treinamento D se e somente se ela classifica corretamente estes exemplos.

Eliminação do Candidatos

- O subespaço de todas hipóteses é chamado, **Espaço Versão** (*version space*) com respeito ao espaço de hipóteses H e os exemplos de treinamento D , porque ele contém todas as versões plausíveis do conceito alvo.
- Para representar o Espaço Versão, listamos todos os seus membros e eliminamos aqueles que não satisfazem um critério.
- Pode ser aplicado sempre que o espaço de hipóteses for finito.
- É garantido que ele forneça todas as hipóteses consistentes com os dados de treinamento.

Eliminação do Candidatos

Algoritmo

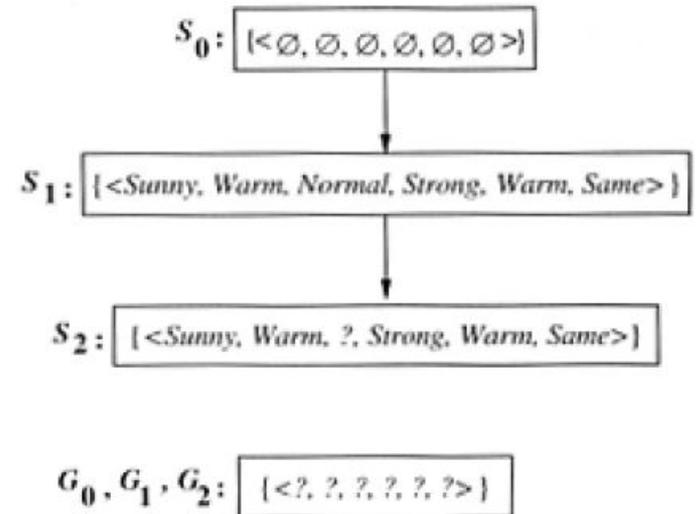
- O algoritmo Eliminação de Candidatos representa o Espaço Versão armazenando somente seus membros mais gerais (G) e mais específicos (S)
- Dado somente G e S,
 - é possível enumerar todos os membros do Espaço Versão.

Eliminação do Candidatos

Exemplo

- Passo 1:
 - $S_0 = [\emptyset, \emptyset, \emptyset, \emptyset, \emptyset, \emptyset]$
 - $G_0 = [?, ?, ?, ?, ?, ?]$
- Estas duas fronteiras delimitam o espaço de hipótese inteiro.
- S_0 e G_0 são as fronteiras iniciais que correspondem as hipótese mais específicas e gerais, respectivamente

Eliminação do Candidatos



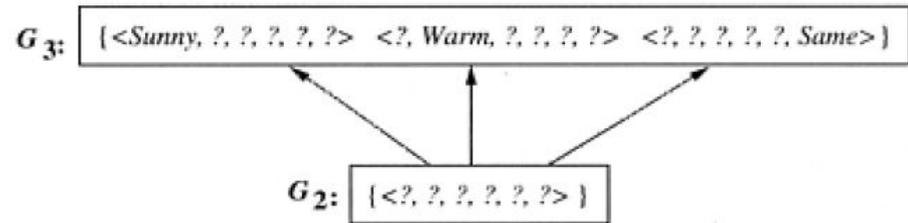
Exemplos de Aprendizagem

1. $[\text{Sunny}, \text{Warm}, \text{Normal}, \text{Strong}, \text{Warm}, \text{Same}] = \text{YES}$
2. $[\text{Sunny}, \text{Warm}, \text{High}, \text{Strong}, \text{Warm}, \text{Same}] = \text{YES}$

- Primeira Iteração:
 - Os exemplos de aprendizagem 1 e 2 tornam a fronteira S mais geral, assim como no algoritmo Find-S.
 - Entretanto, não tem efeito em G

Eliminação do Candidatos

S_2, S_3 : { <Sunny, Warm, ?, Strong, Warm, Same> }

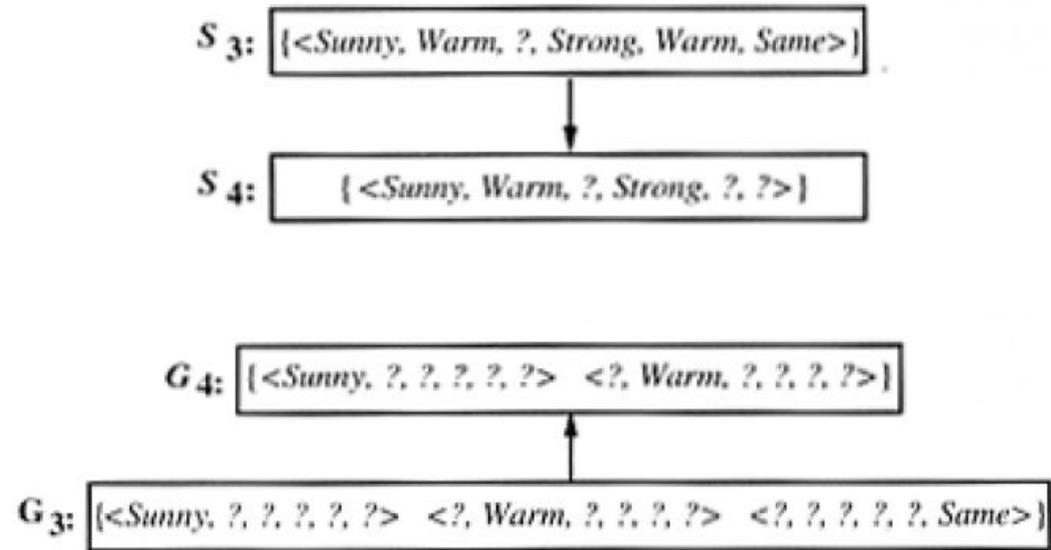


Exemplo de Aprendizagem

1. [Rainy, Cold, High, Strong, Warm, Change] = No

- Segunda Iteração:
 - O terceiro exemplo de aprendizagem é negativo, o que força a fronteira G_2 se tornar mais especializada, gerando G_3 .
 - Note que **diversas** hipóteses válidas são incluídas em G_3 .

Eliminação do Candidatos



Exemplo de Aprendizagem

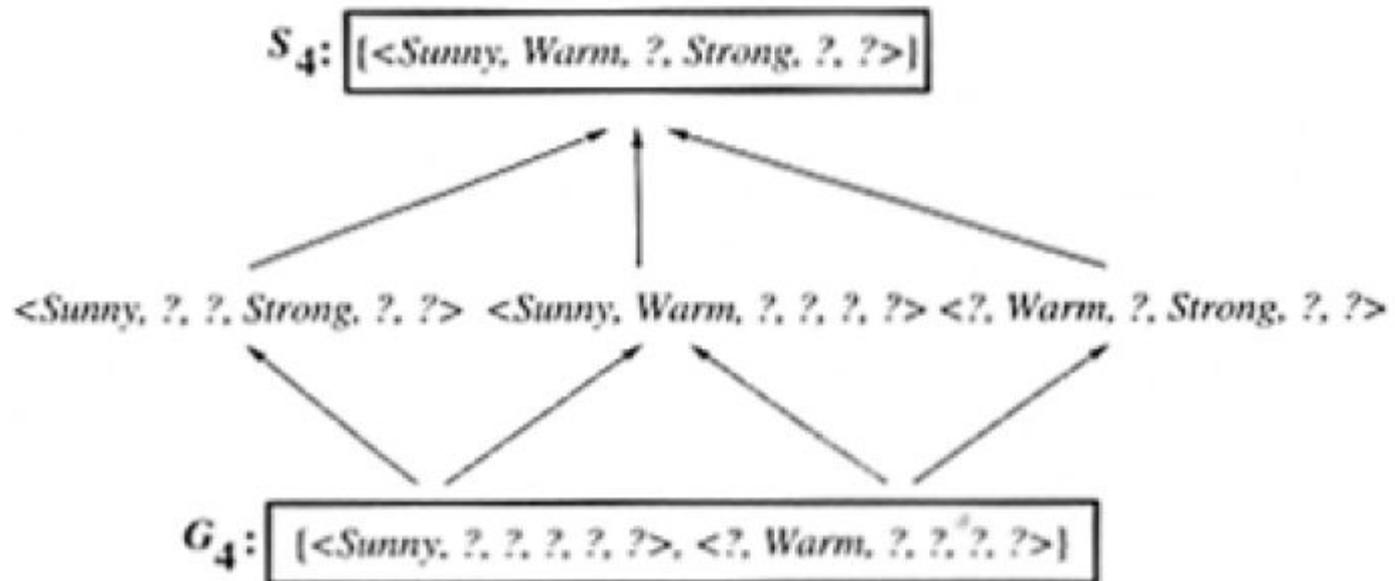
1. [Sunny, Warm, High, Strong, Cool, Change] = YES

- Terceira Iteração:
 - O quarto exemplo de aprendizagem generaliza a fronteira S , gerando S_4 .
 - Um membro de G_3 deve ser eliminado, gerando G_4 .

Eliminação do Candidatos

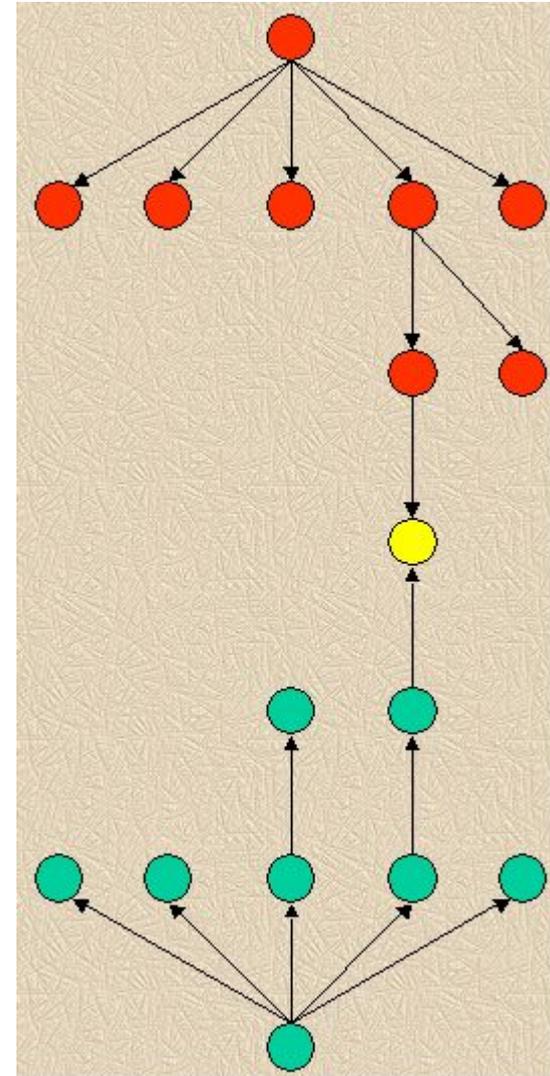
O Espaço de Versão final para nosso problema exemplo.

- O espaço aprendido é independente da sequência na qual os exemplos de treinamento são apresentados.
- Se mais dados de treinamento forem usados, as fronteiras S e G se moverão monotonicamente mais perto umas das outras, delimitando um Espaço Versão de Hipóteses Candidatas cada vez menor.



Eliminação do Candidatos

- Eventualmente, exemplos positivos e negativos podem convergir para uma solução comum.



Eliminação do Candidatos

Classificando Novas Hipóteses

Como utilizar então o Espaço de Versão para classificar novas instâncias?

Instance	<i>Sky</i>	<i>AirTemp</i>	<i>Humidity</i>	<i>Wind</i>	<i>Water</i>	<i>Forecast</i>	<i>EnjoySport</i>
A	Sunny	Warm	Normal	Strong	Cool	Change	?
B	Rainy	Cold	Normal	Light	Warm	Same	?
C	Sunny	Warm	Normal	Light	Warm	Same	?
D	Sunny	Cold	Normal	Strong	Warm	Same	?

- A instância A é classificada como positiva por todas as hipóteses no Espaço Versão final
- A instância B é classificada como negativa por todas as hipóteses no Espaço Versão final

Eliminação do Candidatos

Classificando Novas Hipóteses

Como utilizar então o Espaço de Versão para classificar novas instâncias?

Instance	<i>Sky</i>	<i>AirTemp</i>	<i>Humidity</i>	<i>Wind</i>	<i>Water</i>	<i>Forecast</i>	<i>EnjoySport</i>
A	Sunny	Warm	Normal	Strong	Cool	Change	?
B	Rainy	Cold	Normal	Light	Warm	Same	?
C	Sunny	Warm	Normal	Light	Warm	Same	?
D	Sunny	Cold	Normal	Strong	Warm	Same	?

- A instância C apresenta uma situação diferente. Metade das hipótese dos espaço versão classificam como positiva e metade como negativa.
- **Atenção:** Não se pode classificar a instância C com confiança. A mesma deve ser **rejeitada**

Eliminação do Candidatos

Classificando Novas Hipóteses

Como utilizar então o Espaço de Versão para classificar novas instâncias?

Instance	<i>Sky</i>	<i>AirTemp</i>	<i>Humidity</i>	<i>Wind</i>	<i>Water</i>	<i>Forecast</i>	<i>EnjoySport</i>
A	Sunny	Warm	Normal	Strong	Cool	Change	?
B	Rainy	Cold	Normal	Light	Warm	Same	?
C	Sunny	Warm	Normal	Light	Warm	Same	?
D	Sunny	Cold	Normal	Strong	Warm	Same	?

- A instância D é classificada como positiva por duas hipóteses do Espaço Versão e como negativa por outras quatro hipótese do E.V.
- Se levarmos em conta a maioria, D deve ser classificada como negativa.
- Porém não é uma decisão unânime

Resumo

- Aprendizagem de conceito pode ser vista como um problema de busca em um espaço pré-definido de hipóteses em potencial.
- A ordenação parcial “geral a específico” de hipóteses fornece uma estrutura útil para organizar a busca no espaço de hipóteses.

Resumo

- O algoritmo **Find-S** utiliza este ordenamento “geral a específico”, realizando uma busca de específica a geral no espaço de hipóteses para encontrar a hipótese mais específica consistente com os exemplos de treinamento.
- O algoritmo Eliminação de Candidatos utiliza este ordenamento parcial “geral a específico” para calcular o **Espaço Versão** através da computação incremental dos conjuntos mais específicos (S) e mais gerais (G).
- As aplicações práticas são limitadas pelo fato de que os dois algoritmos têm um desempenho fraco na presença de dados de treinamento com **ruído**.

Exercícios

- Execute os algoritmos:
 - Find-S
 - Candidate Elimination

para o conceito abaixo

Size	Color	Shape	Label
big	red	circle	No-
small	red	triangle	No-
small	red	circle	Yes+
big	blue	circle	No-
small	blue	circle	Yes+

Referências

- Dr. Ilyas Cicekli, **Concept Learning**,
<https://web.cs.hacettepe.edu.tr/~ilyas/Courses/BIL712/lec01-conceptLearning.pdf>
- Luiz E. S Oliviera, **Aprendizagem de Conceito**, DIInf / UFPR.
- Stanford Scholar,
Practical Machine Learning: 2.5 - Concept Learning
<https://www.youtube.com/watch?v=2C4GiluGkSY>
- Stanford Scholar,
Practical Machine Learning: 2.6 - General to Specific Ordering,
https://www.youtube.com/watch?v=Nhu_7OeZRvU