기계 학습의 기초

(Rule-based Machine Learning / Decision Tree / Linear Regression)

특허법인 가산

특허 6팀

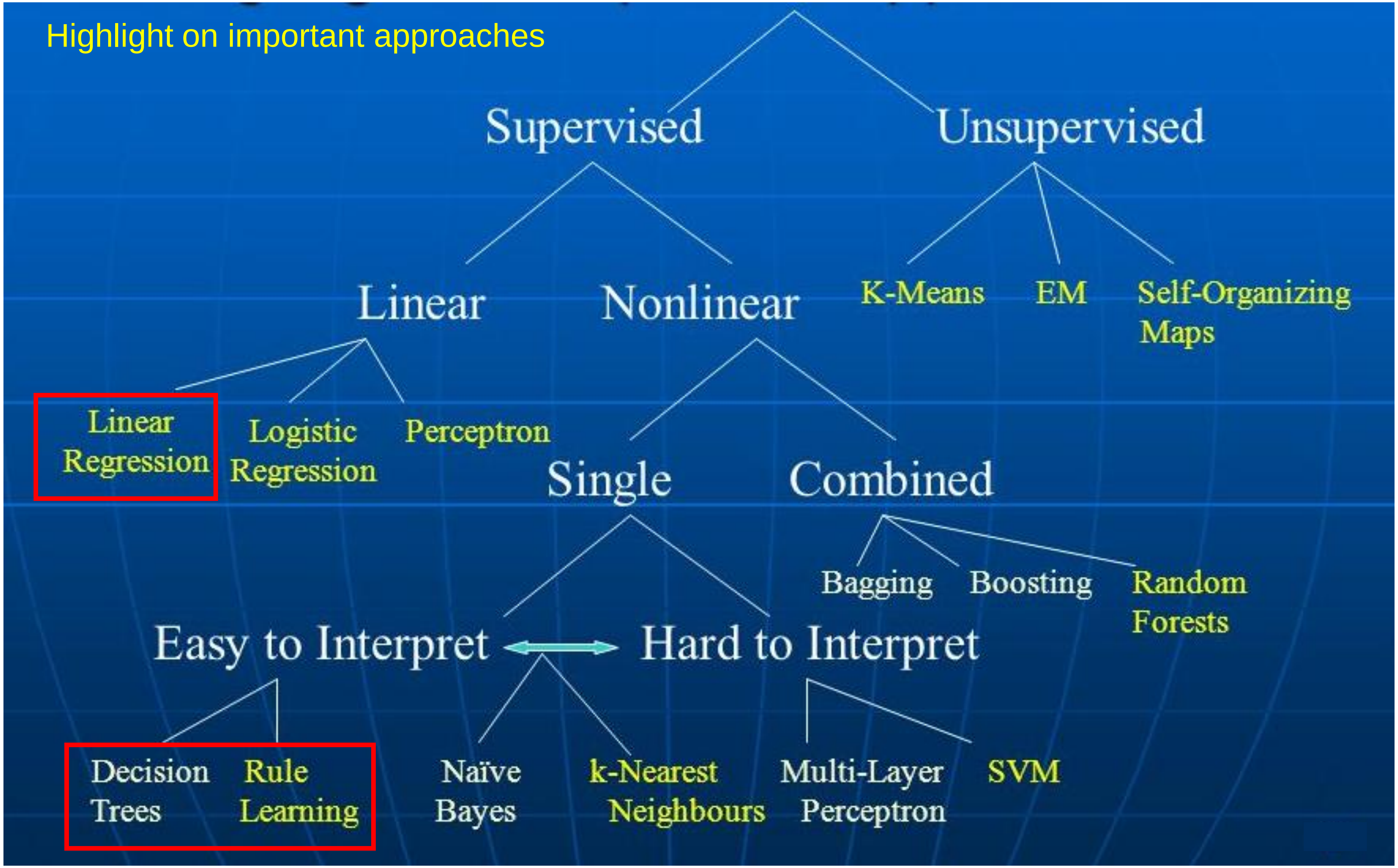
박병규

목 차

KASAN
on your side

1. Overview

Machine Learning Taxonomy



2. Rule-based Machine Learning

A Perfect World for Rule Based ML

KASAN
on your side

◆ Perfect world

- ➢ Training Data
  - ▪ Error-free, Noise-free

  → No observation error, No inconsistent observation
- ➢ Target Function
  - ▪ Deterministic

  → No stochastic elements
  - ▪ Contained in hypotheses set

  → Full information to regenerate the system

◆ Real world

- ➢ Noise
- ➢ Inconsistence

FA(Function Approximation)

◆ Example : EnjoySport

Feature O

Label Y

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

Training dataset D

Instance X

Target Function C
C(x) : X→Y

Approximation

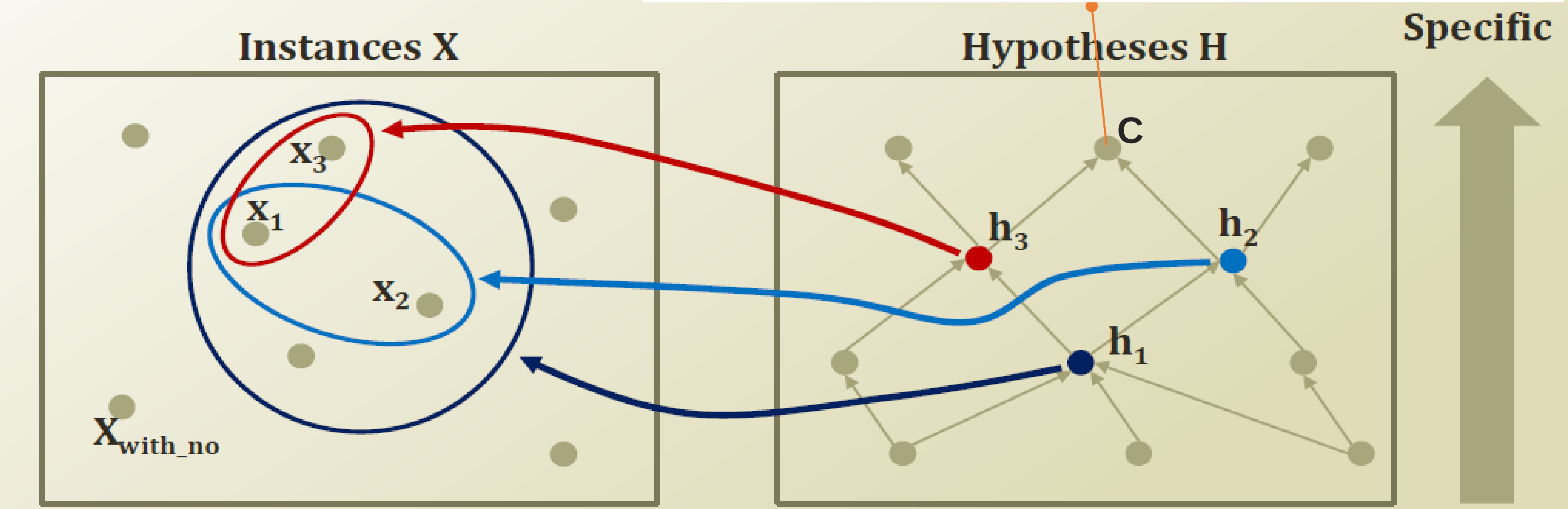
Hypothesis H
H(x) : X→Y in D

Graphical Representation of FA

◆ Hypotheses H

- conjunction of constraints on features

Assumption : Hypotheses Space H includes Target Function C!



x_1: <Sunny, Warm, Normal, Strong, Warm, Same>

x_2: <Sunny, Warm, Normal, Light, Warm, Same>

x_3: <Sunny, Warm, Normal, Strong, Warm, Change>

h_1: <Sunny, ?, ?, ?, Warm, ?>

h_2: <Sunny, ?, ?, ?, Warm, Same>

h_3: <Sunny, ?, ?, Strong, Warm, ?>

Find-S Algorithm

◆ Find-S

- Objective :- Finding a Maximally Specific Hypothesis
- Method :- Search Hypotheses Space in General-to-Specific order

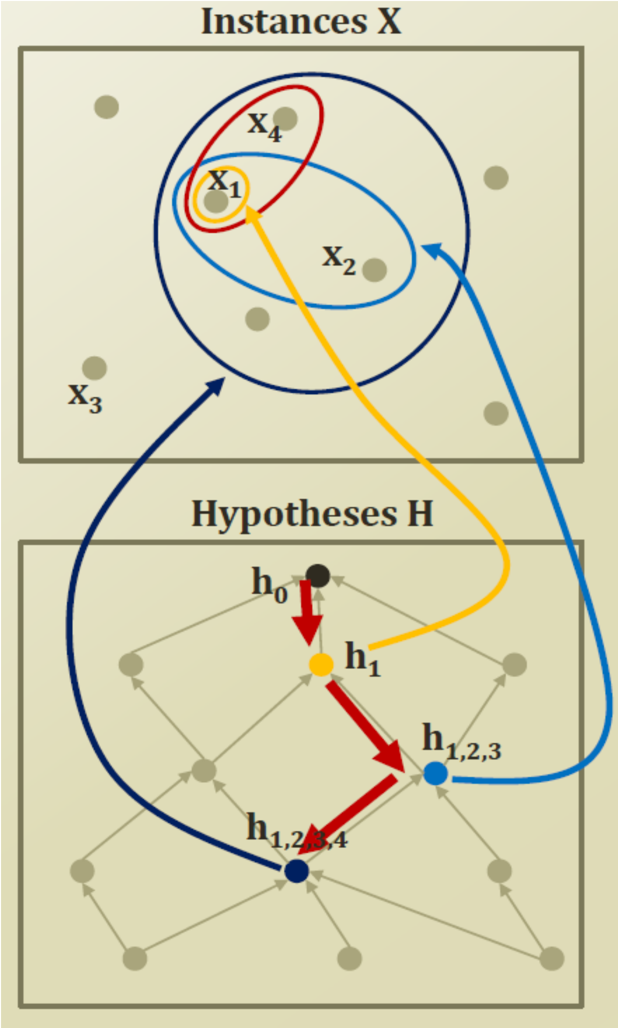
```
Initialize h to the most specific in H   // < Ø, Ø, Ø, Ø, Ø>

FOR instance x in D {
      IF x is positive {
            FOR feature f in O
                  IF fi in h == fi in x
                        Do nothing
                  ELSE
                        fi in h = fi in h ∪ fi in x
      }
}
RETURN h
```

Find-S Algorithm (1/2)

◆ Example

Step	Instance x	Hypotheses h
0	-	< Ø, Ø, Ø, Ø, Ø>
1	x_1 : <Sunny, Warm, Normal, Strong, Warm, Same>	h_1 : <Sunny, Warm, Normal, Strong, Warm, Same>
2	x_2 : <Sunny, Warm, Normal, Light, Warm, Same>	$h_{1,2}$:<Sunny, Warm, Normal, ?, Warm, Same>
3	x_3 : negative → do nothing	$h_{1,2,3}$:<Sunny, Warm, Normal, ?, Warm, Same>
4	x_4 : <Sunny, Warm, Normal, Strong, Warm, Change>	$h_{1,2,3,4}$:<Sunny, Warm, Normal, ?, Warm, ?>



Find-S Algorithm (2/2)

◆ Properties

- Ignores every negative example
- Guaranteed to output the most specific hypothesis consistent with the positive training examples (for conjunctive hypothesis space)

◆ Problems

- Converged to the correct target concept ?

  → No way to know whether the solution is unique
- Why prefer the most specific hypothesis?

  How about the most general hypothesis?

◆ Version Space

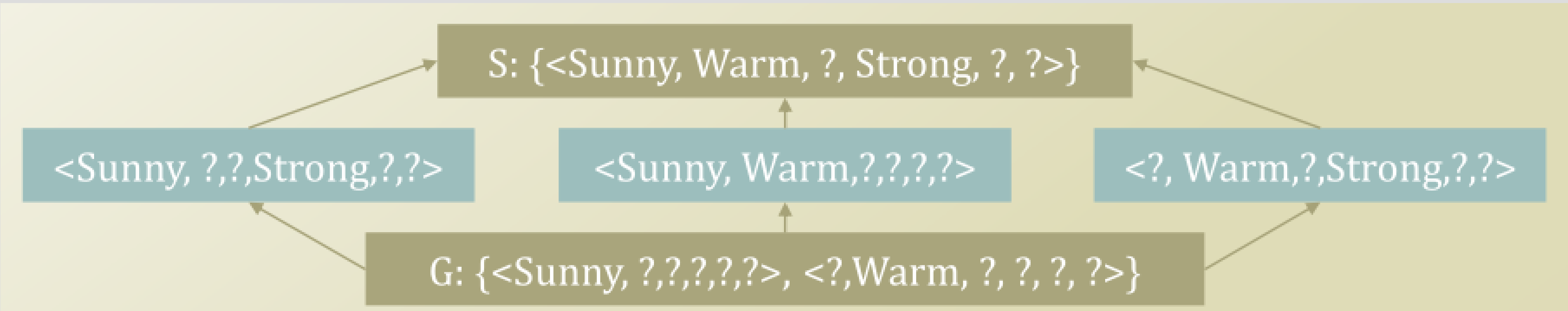
- Many hypotheses possible, and No way to find the convergence

  → Setup the perimeter of the possible hypotheses

- Definition :- set of the possible hypotheses
  - General Boundary G , Specific Boundary, S
  - Every hypothesis, h satisfies

$$\mathrm{VS}_{\mathrm{H,D}} = \{\ h \in H \mid \exists\, s \in S,\ \exists\, g \in G,\ \ g \geq h \geq s\ \}$$

where $x \geq y$ means x is more general or equal to y



CEA(Candidate Elimination Algorithm)

```
Initialize S to maximally specific h in H  // S0 : {< Ø, Ø, Ø, Ø, Ø>}
Initialize G to maximally general h in H   // G0: {<?, ?, ?, ?, ?, ?>}

FOR instance x in D
    (A) IF y of x is positive  // special → general
     (A-1)  Generalize S as much as needed to cover o in x
     (A-2)  Remove any h in G, for which h(o)≠ y
    (B) IF y of x is negative  // general → special
     (B-1)  Specialize G as much as needed to exclude o in x
     (B-2)  Remove any h in S, for which h(o)= y

Generate h that satisfies ∃s∈S, ∃g∈G, g ≥ h ≥ s
```

CEA Examples (1/4)

	Sky	Temp	Humid	Wind	Water	Forecst	EnjoySpt
X_1 →	Sunny	Warm	Normal	Strong	Warm	Same	Yes
X_2 →	Sunny	Warm	High	Strong	Warm	Same	Yes
X_3 →	Rainy	Cold	High	Strong	Warm	Change	No
X_4 →	Sunny	Warm	High	Strong	Cool	Change	Yes

S0: {<Ø, Ø, Ø, Ø, Ø, Ø>}

(A-1) ↓ X_1

S1: {<Sunny, Warm, Normal, Strong, Warm, Same>}

(A-1) ↓ X_2

S2: {<Sunny, Warm, ?, Strong, Warm, Same>}

G0, G1, G2: {<?,?,?,?,?,?>}

CEA Examples (2/4)

	Sky	Temp	Humid	Wind	Water	Forecst	EnjoySpt
X_1 →	Sunny	Warm	Normal	Strong	Warm	Same	Yes
X_2 →	Sunny	Warm	High	Strong	Warm	Same	Yes
X_3 →	Rainy	Cold	High	Strong	Warm	Change	No
X_4 →	Sunny	Warm	High	Strong	Cool	Change	Yes

S0: {<Ø, Ø, Ø, Ø, Ø, Ø>}

(A-1) ↓ X_1

S1: {<Sunny, Warm, Normal, Strong, Warm, Same>}

(A-1) ↓ X_2

S2, S3: {<Sunny, Warm, ?, Strong, Warm, Same>}

G3: {<Sunny,?,?,?,?,?>, <?,Warm,?,?,?,?>, <?,?,?,?,?,Same>}

(B-1) ↑ X_3

G0, G1, G2: {<?,?,?,?,?,?>}

CEA Examples (3/4)

	Sky	Temp	Humid	Wind	Water	Forecst	EnjoySpt
X_1 →	Sunny	Warm	Normal	Strong	Warm	Same	Yes
X_2 →	Sunny	Warm	High	Strong	Warm	Same	Yes
X_3 →	Rainy	Cold	High	Strong	Warm	Change	No
X_4 →	Sunny	Warm	High	Strong	Cool	Change	Yes

S0: {<Ø, Ø, Ø, Ø, Ø, Ø>}

(A-1) ↓ X_1

S1: {<Sunny, Warm, Normal, Strong, Warm, Same>}

(A-1) ↓ X_2

S2, S3: {<Sunny, Warm, ?, Strong, Warm, Same>}

(A-1) ↓ X_4

S4: {<Sunny, Warm, ?, Strong, ?, ?>}

Still many ***h*** s

G4: {<Sunny,?,?,?,?,?>, <?,Warm,?,?,?,?>}

(B-2) ↑ X_4

G3: {<Sunny,?,?,?,?,?>, <?,Warm,?,?,?,?>, <?,?,?,?,?,Same>}

(B-1) ↑ X_3

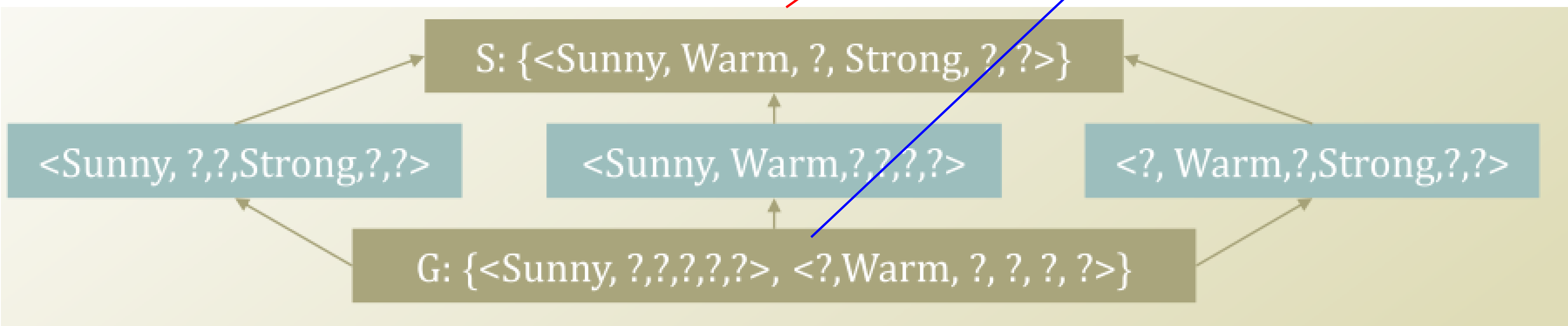
G0, G1, G2: {<?,?,?,?,?,?>}

conflict

CEA Examples (4/4)

◆ How to classify

No.	Instance x	Classification Result
1	<Sunny, Warm, Normal, Strong, Cool, Change>	Y
2	<Rainy, Cold, Normal, Light, Warm, Same>	N
3	<Sunny, Warm, Normal, $\boxed{\text{Light,}}$ Warm, Same>	??? (→need more experience)



<Version Space>

Pros and Cons or Limitations of CEA

◆ Pros and Cons

- Don't have to store in memory every rule consistent with the examples - only the S and G sets
- Performs an exhaustive search of the space of all possible classification rules
- Not robust to any noise

◆ Limitations

- Will the candidate elimination algorithm converge to the correct hypothesis?
  - Given the assumption of the Perfect World
  - Converge? → Yes
  - Correct? → Yes
- Correct h can be removed by the noise in the Real World

◆ DNA분자를 이용한 개념 학습의 구현방법

- 출원번호 : 10-2002-0048910
- 상태 : 거절 확정

[청구항 1]
개념 학습의 과정을, **가설 및 학습 데이터의 예가 속성 값들의 연언(and)으로 표현**되고, 개념이 될 수 있는 **가능한 모든 가설들의 집단인 가설공간(version space)으로부터 긍정예 또는 부정예로서의 학습데이터의 입력에 의해 모순되지 않은 가설들을 선택**해내고, 이러한 갱신 과정을 반복함으로써, 목표 개념에 해당하는 가설을 탐색하는 과정으로 규정하고 이러한 학습을 DNA 컴퓨팅을 이용하여 구현함에 있어서,
a) 상기 속성 값을, 속성마다 동일한 스티키 말단(sticky end)의 서열 및 속성 값마다 다른 이중가닥 서열을 갖는 이중가닥의 DNA에 대응시키고,
b) 상기 속성에 대응되는 이중가닥 DNA를 임의 결찰(ligation)하여, 초기의 전체 가설공간을 형성하며,
c) 긍정예 또는 부정예로서의 학습데이터의 입력에 의해 모순되지 않은 가설들을 비드를 이용한 친화분리 방법에 의해 선택해 내고,
d) 상기 c)의 과정을 반복함으로써, 목표 개념에 해당하는 DNA를 수득함을 특징으로 하는, DNA 컴퓨팅을 이용한 **개념학습**의 구현방법.

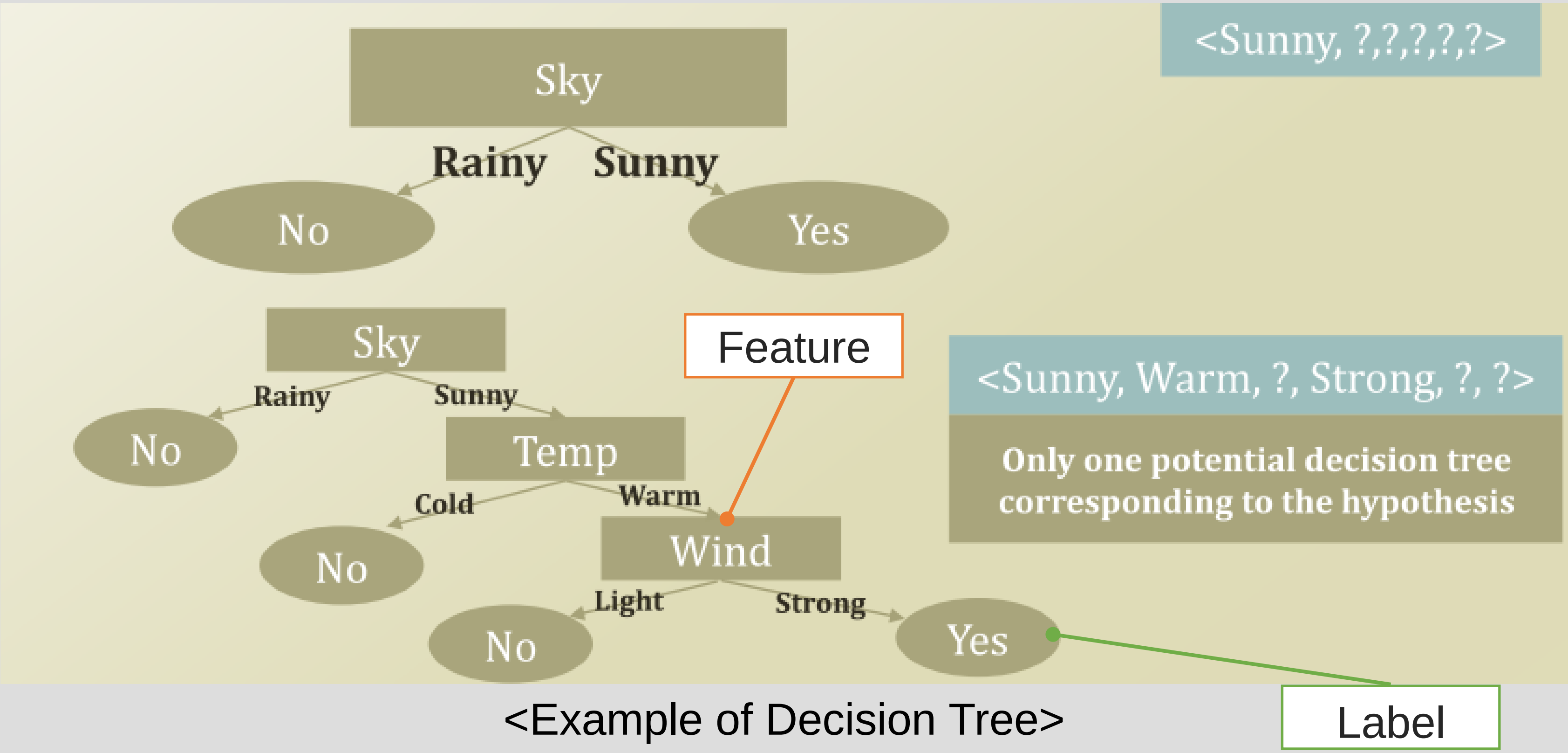
3. Decision Tree

Motivation

◆ Decision Tree

➢ Need a better learning method

- More robust to noise



<Example of Decision Tree>

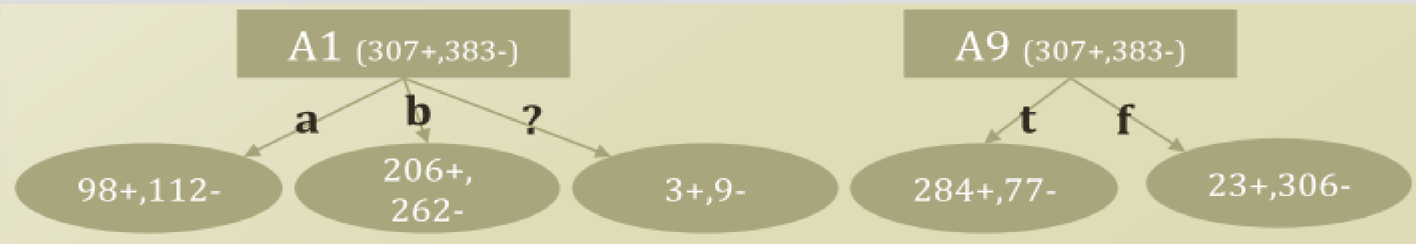
Training Dataset

◆ Credit Approval

➢ http://archive.ics.uci.edu/ml/datasets/Credit+Approval

A1: b, a.
A2: continuous.
A3: continuous.
A4: u, y, l, t.
A5: g, p, gg.
A6: c, d, cc, i, j, k, m, r, q, w, x, e, aa, ff.
A7: v, h, bb, j, n, z, dd, ff, o.
A8: continuous.
A9: t, f.
A10: t, f.
A11: continuous.
A12: t, f.
A13: g, p, s.
A14: continuous.
A15: continuous.
C: +,- (class attribute)

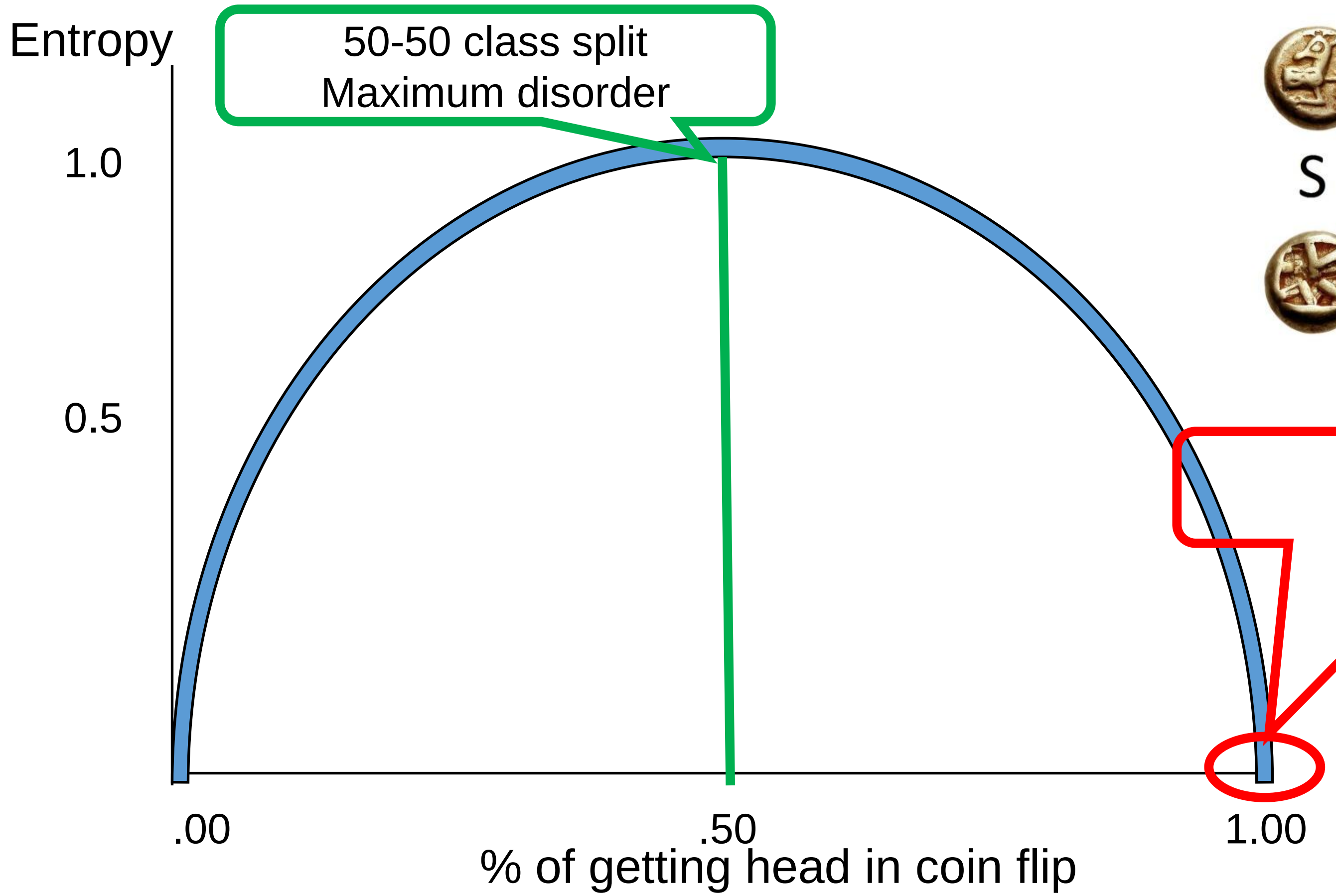
- 690 instances total
  - ✓ 307 positive instances
  - ✓ 383 negative instances
- Considering A1
  - ✓ 98 positive when a
  - ✓ 112 negative when a
  - ✓ 206 positive when b
  - ✓ 262 negative when b
  - ✓ 3 positive when ?
  - ✓ 9 negative when ?
- Considering A9
  - ✓ 284 positive when t
  - ✓ 77 negative when t
  - ✓ 23 positive when f
  - ✓ 306 negative when f

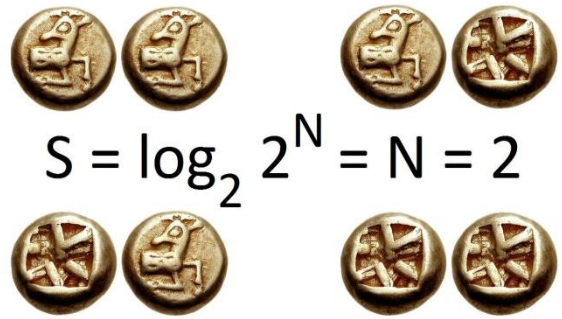


Shannon Entropy (1/3)

◆ Entropy

- Measure of unpredictability(or uncertainty) of information content
- Measure of the average amount of information

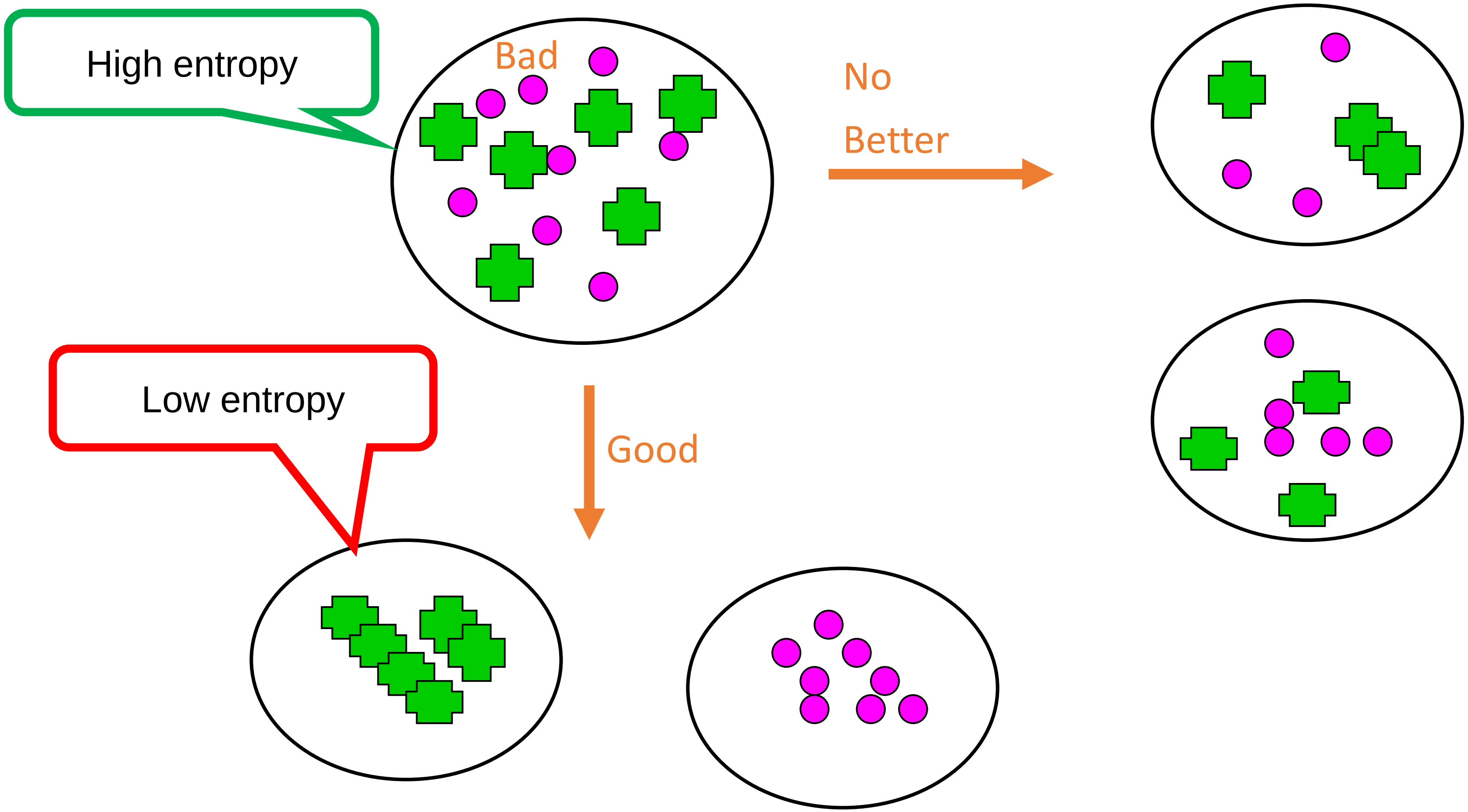




$$S = \log_2 2^N = N = 2$$

Shannon Entropy (2/3)

◆ Entropy



Shannon Entropy (3/3)

◆ Entropy of a Random Variable

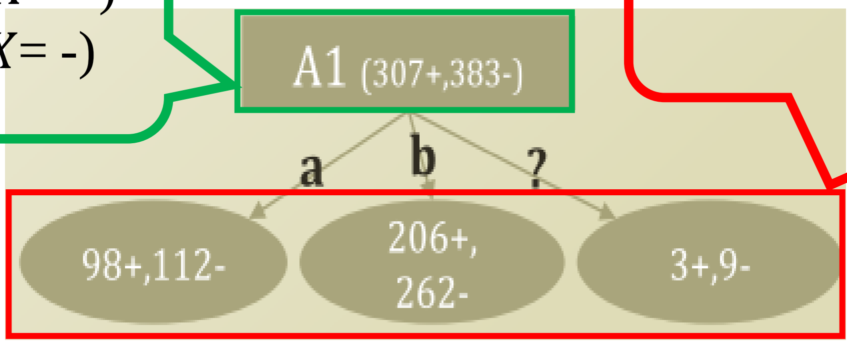
✓ $H(X) = -\sum_{x} P(X)\ log_b P(X = x)$

◆ Conditional Entropy

✓ $H(Y|X) = \sum_{x} P(X = x) H(Y|X = x)$

$= \sum_{x} P(X = x)\{-\sum_{y} P(Y = y|X = x) log_b P(Y = y|X = x)\}$

$$H(X) = -P(X = +)\ log_b P(X = +) - P(X = -)\ log_b P(X = -)$$



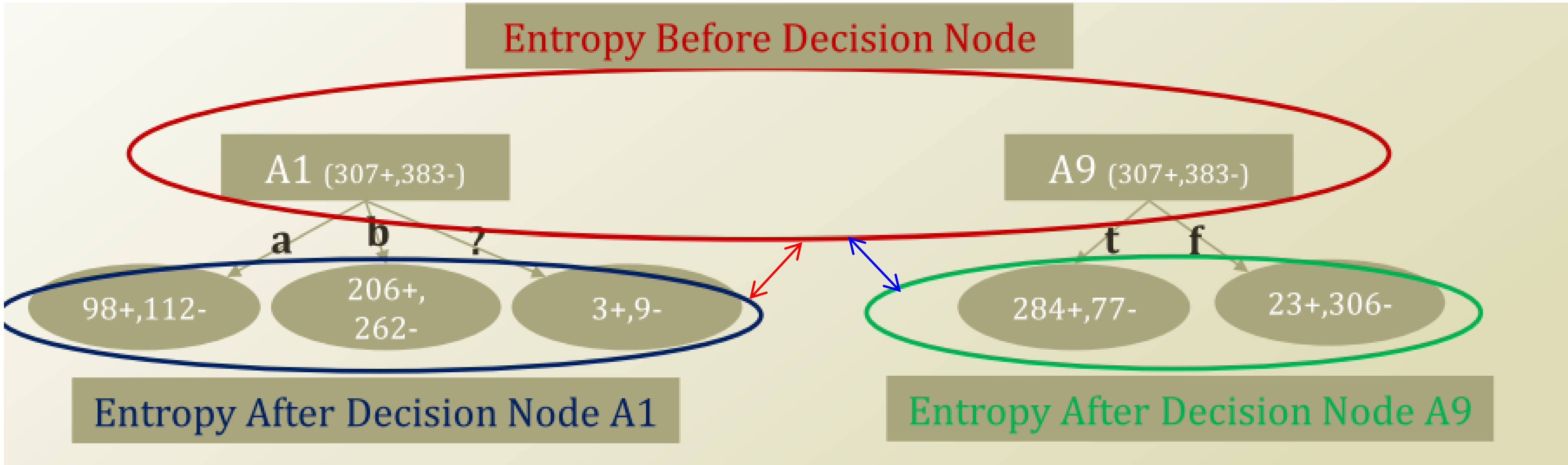
$$H(Y|A1) = \sum_{A1=a,b,?} \{-P(Y = +\ |\ A1) log_b P(Y=+|A1) - P(Y = -\ |\ A1) log_b P(Y=-|A1)\}$$

IG(Information Gain)

◆ IG

- Measure of expected *reduction* in entropy

✓ $IG(Y, A_i) = H(Y) - H(Y|A_i)$



IG(Y, A1) < IG(Y, A2)

Top-Down Induction Algorithm (1/3)

◆ Variations

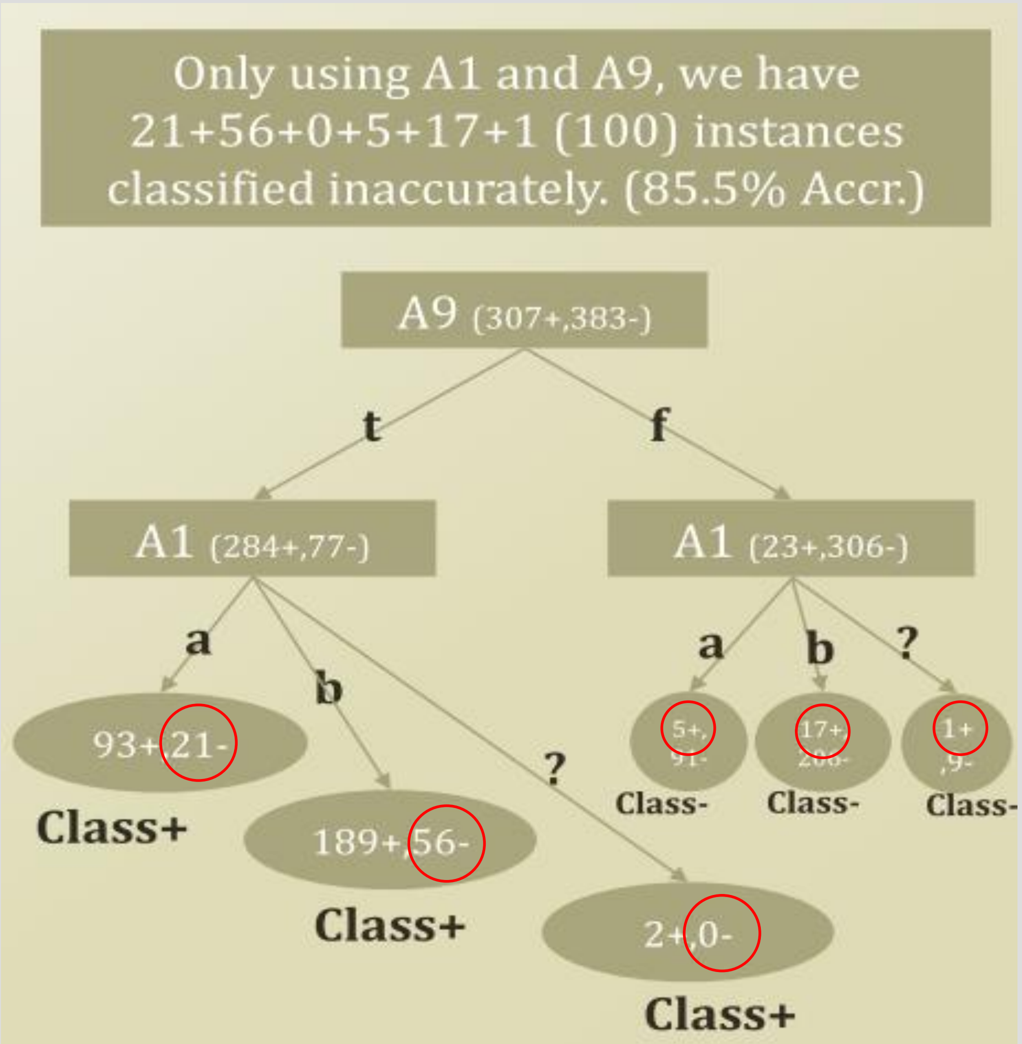
- TDIDT(Top Down Induction of Decision Trees)
- ID3(Iterative Dichotomiser 3), CART..

◆ ID3 Algorithm

```
Create an initial open node
Put instances in the initial node
Repeat until no open node
    Select an open node to split
    Select a best variable to split    // utilize IG, greedy approach
    FOR values of the selected variable
        Sort instances with the value of the selected variable
        Put the sorted items under the branch of the value of the
        variable
        IF the sorted items are all in one Class
            Close the leaf node of the branch
```

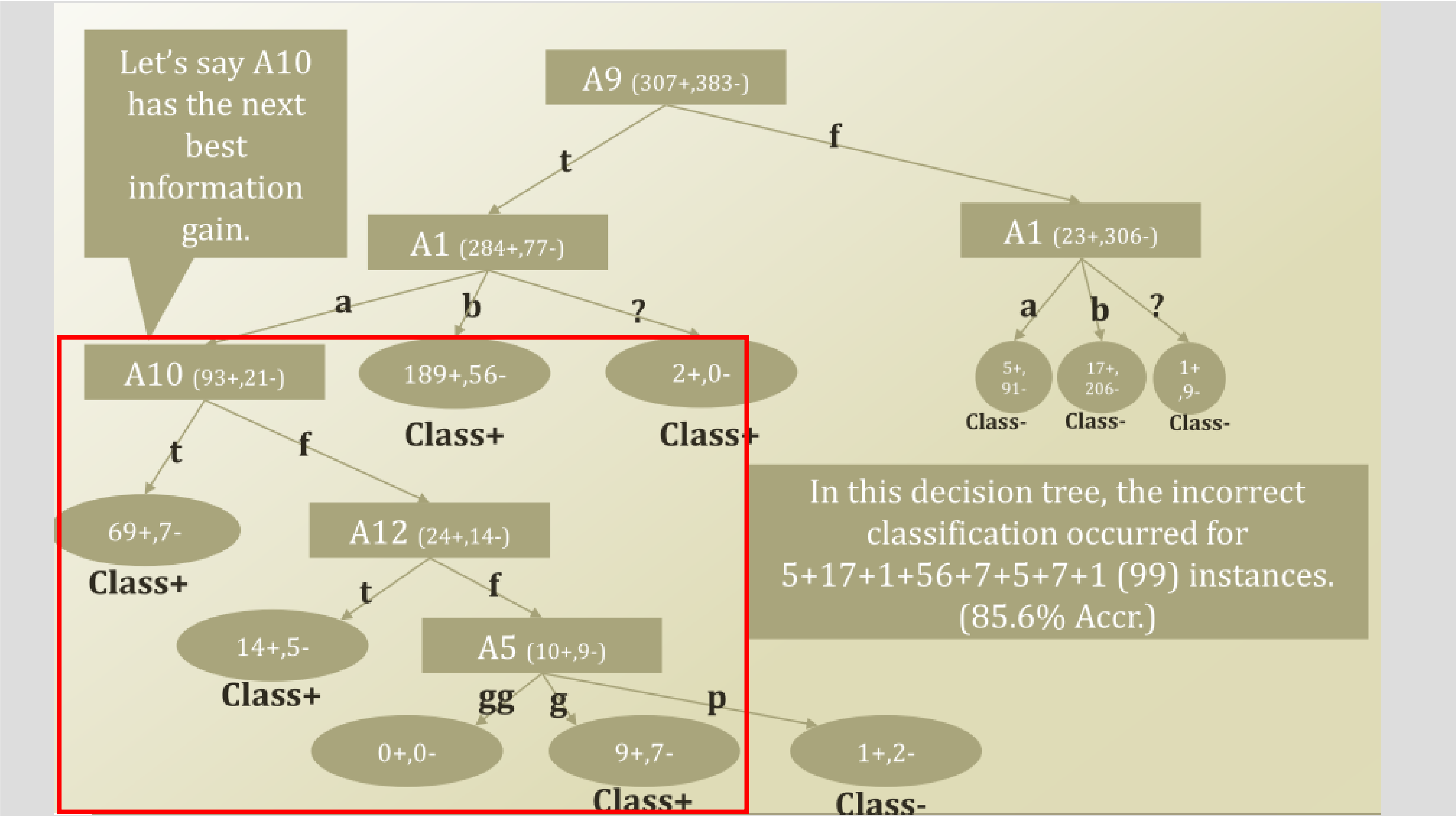
Top-Down Induction Algorithm (2/3)

◆ Result of ID3 Algorithm



Top-Down Induction Algorithm (3/3)

◆ Result of ID3 Algorithm



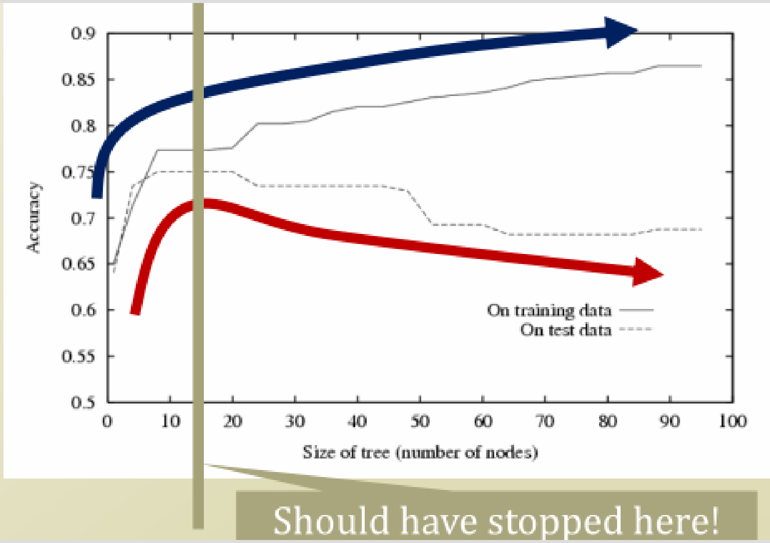
Pros and Cons or Limitations of DT

◆ Pros and Cons

- Robust to noise
- Easy to interpret
- Prone to overfitting, need pruning techniques

◆ Limitations

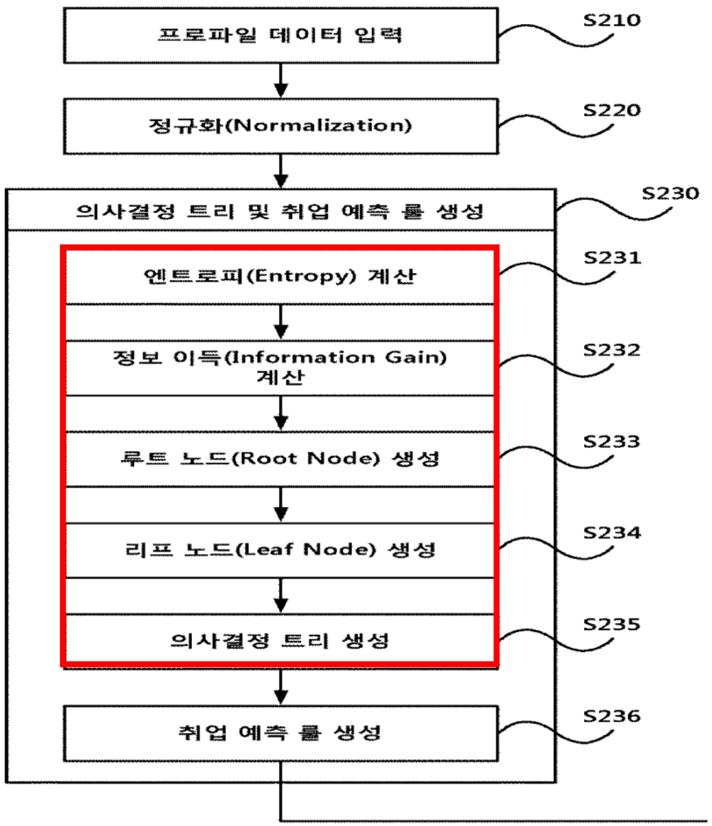
- DT is overfit when exists another DT’ and
  - DT has smaller error on training examples, but
  - DT has bigger error on test examples
- Causes of overfitting
  - Noise
  - Inconsistence
- Solutions
  - Reduced error pruning
  - Early stopping



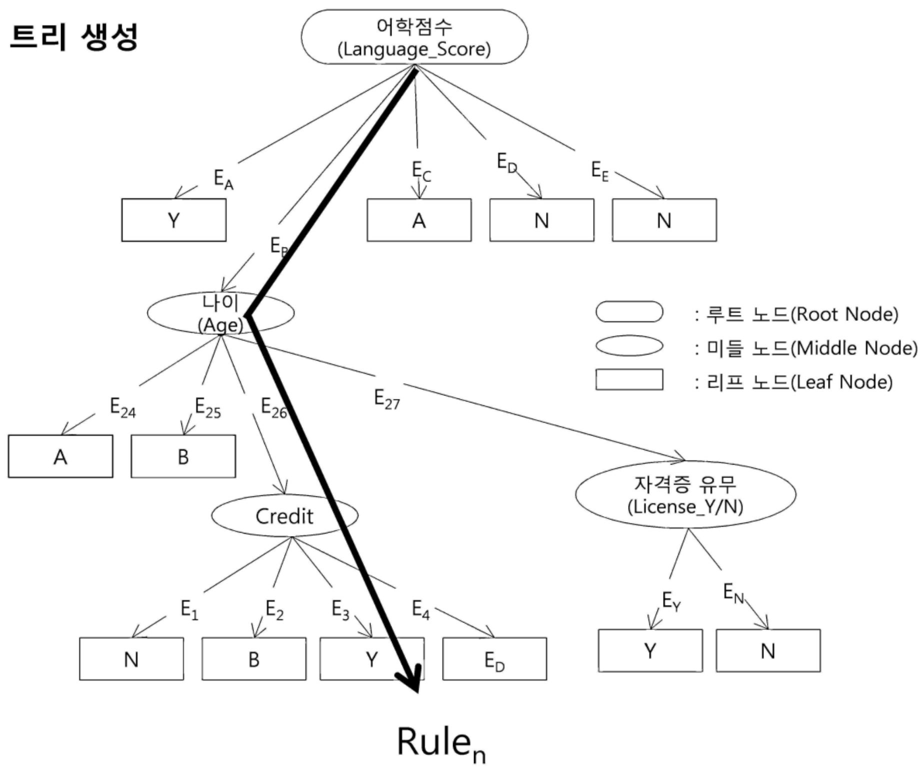
Related Patent

◆ 의사결정 트리를 이용한 취업 가능성 예측 방법

- 출원번호 : 10-2013-0146803
- 상태 : 등록
- 권리자 : 제주대학교 산학협력단



의사결정 트리 생성



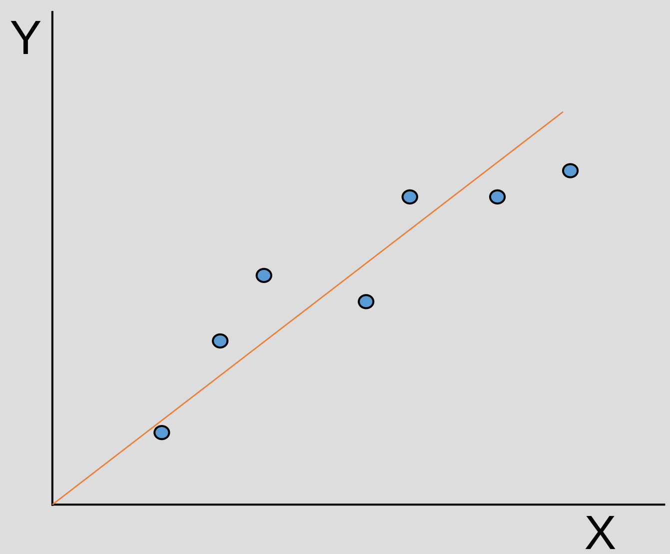
4. Linear Regression

What is Regression?

◆ Regression

◆ Linear Regression

- Predict height from age
- Predict weight price height
- Variable
  - Independent variable X
  - Dependent variable Y



Hypothesis of LR(Linear Regression)

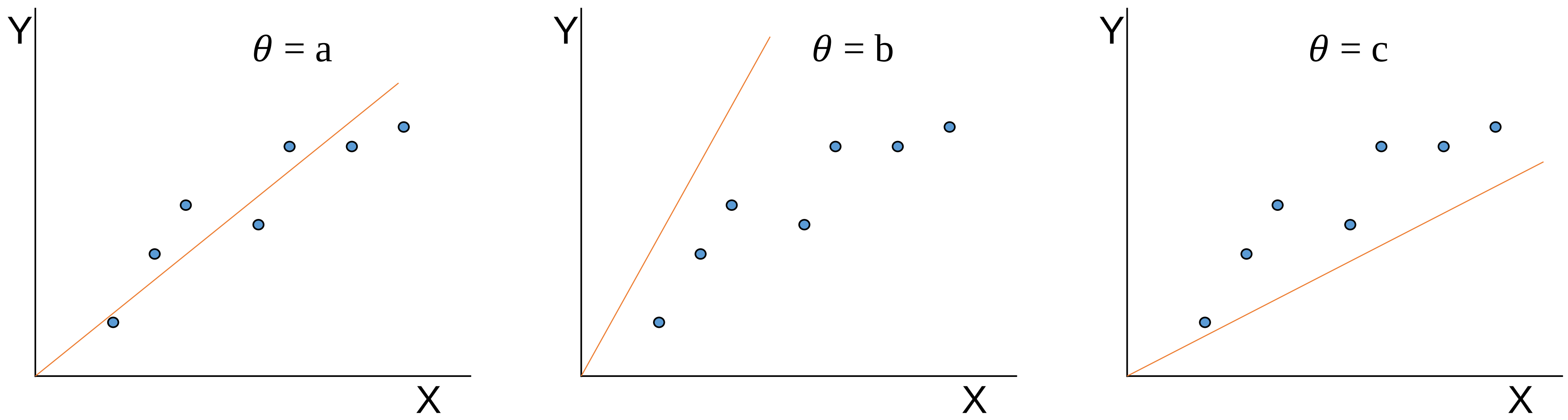
◆ Simple LR

✓ $h : \hat{f}(x;\theta) = \theta_0 + \theta_1 x$

- Linearly weighted sum
  → assumption, can not modify
- Parameter θ
  → approximation factor, need to find better one

◆ Multiple LR

✓ $h : \hat{f}(x;\theta) = \theta_0 + \sum_{\mathrm{i}=1}\theta_i x_i = \sum_{\mathrm{i}=0}\theta_i x_{\mathrm{i}} (x_0 = 1)$

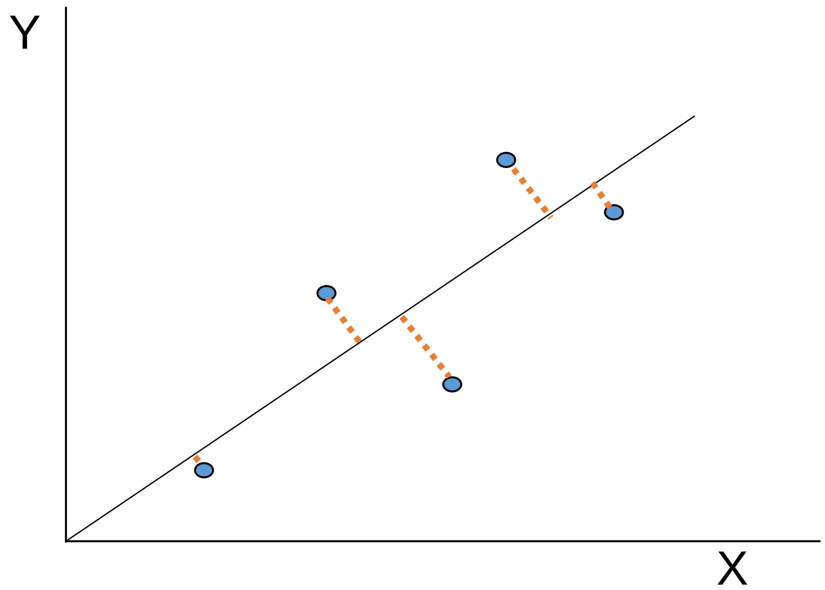


Finding θ in LR (1/3)

◆ Cost function

- Or Loss function
- Represent how close predicted values from hypothesis is to corresponding value
- Utilize least square method

✓ $\mathrm{cost}(\theta) = 1/m \sum_{i=0} (f(x_i;\, \theta) - y^{(i)})^2$



Finding θ in LR (2/3)

◆ Normal Equation

➢ One of methods to find θ minimizing cost

✓ $\theta = (X^T X)^{-1} X^T Y$

➢ Represent x, θ using matrix format

$$h: \hat{f}(x;\theta) = \sum_{i=0}^{n} \theta_i x_i \rightarrow \hat{f} = X\theta$$

$$X = \begin{pmatrix} \boxed{\begin{matrix} 1 \\ \vdots \\ 1 \end{matrix}} & \begin{matrix} \cdots \\ \ddots \\ \cdots \end{matrix} & \begin{matrix} x_1^n \\ \vdots \\ x_n^D \end{matrix} \end{pmatrix}, \theta = \begin{pmatrix} \theta_0 \\ \theta_1 \\ \cdots \\ \theta_n \end{pmatrix}$$

Dummy value

Instance

➢ Find θ that minimizes cost

$$\hat{\theta} = argmin_\theta (f - \hat{f})^2 = argmin_\theta (Y - X\theta)^2$$
$$= argmin_\theta (Y - X\theta)^T (Y - X\theta) = argmin_\theta (Y - X\theta)^T (Y - X\theta)$$
$$= argmin_\theta (\theta^T X^T X\theta - 2\theta^T X^T Y + \boxed{Y^T Y}) = argmin_\theta (\theta^T X^T X\theta - 2\theta^T X^T Y)$$

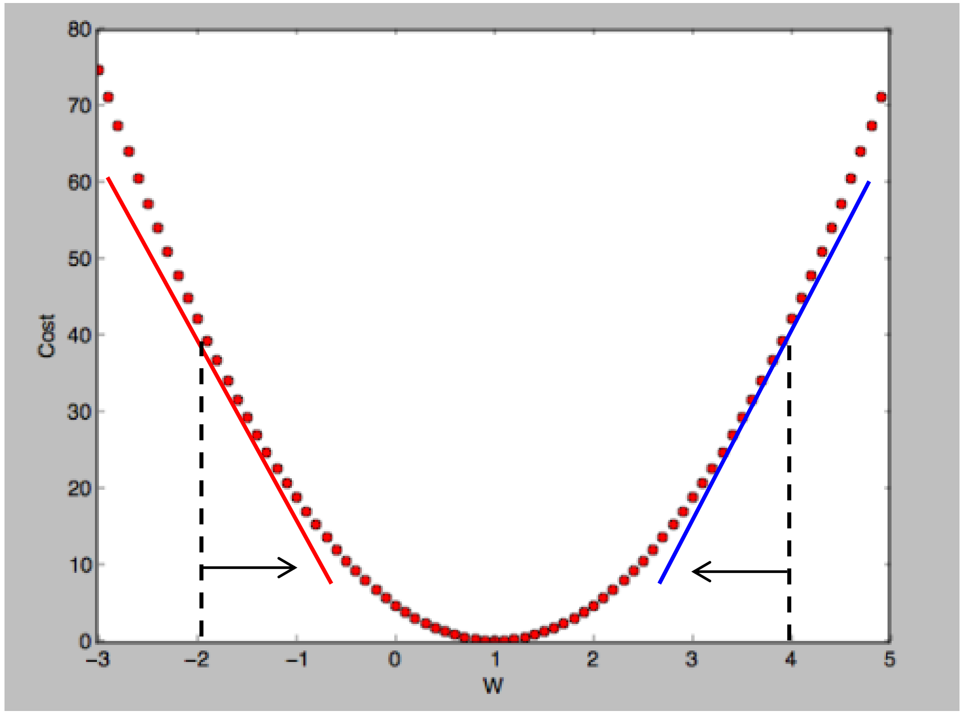
➢ Find θusing derivative

$$\nabla_\theta (\theta^T X^T X\theta - 2\theta^T X^T Y) = 0$$
$$2X^T X\theta - 2X^T Y = 0$$
$$\theta = (X^T X)^{-1} X^T Y$$

◆ Gradient Decent Algorithm

➢ One of methods to find θ minimizing cost

$$cost(W) = \frac{1}{m}\sum_{i=1}^{m}(Wx^{(i)} - y^{(i)})^2$$

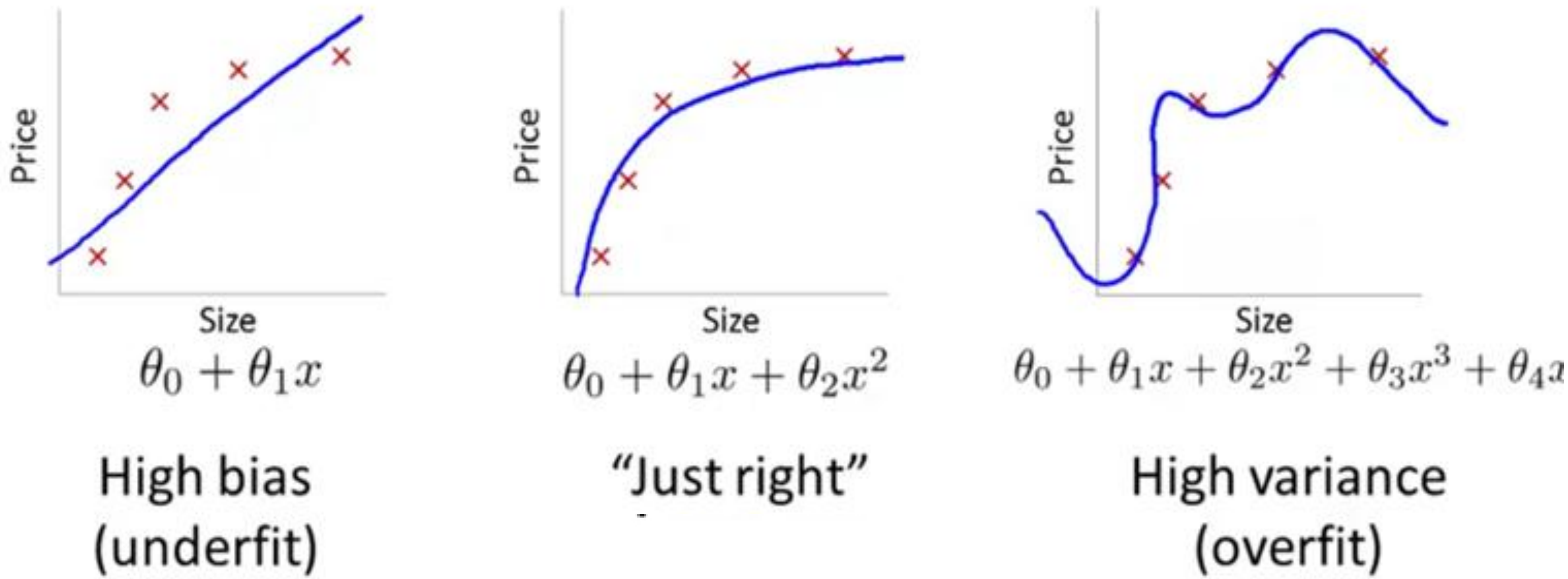


✓ $W := W - \alpha \frac{\partial}{\partial W} cost(W)$

Overfitting problem of LR

◆ Overfitting

- ➢ Similar problem as Decision Tree



Related Patent (1/2)

◆ 장마 시작의 물리 통계 예측 방법

- 출원번호 : 10-2014-0165597
- 상태 : 등록
- 권리자 : 부산대학교 산학협력단

[청구항 1]

장마 시작일에 영향을 미치는 북태평양 고기압의 확장과 이동에 영향을 주는 해수면온도 아노말리를 계산하는 제 1단계;

제 1 단계에서 구한 해수면 온도 아노말리 자료를 이용하여 장마의 시작을 예측하기 위한 **다중선형회귀 예측모델을 구성하는 선행인자를 계산**하는 제 2 단계;

제 2 단계에서 **계산된 선행인자를 구축된 다중선형회귀 예측모델에 대입하여 장마 시작일을 예측**하는 제 3단계;

제 3 단계에서 예측된 장마 시작일을 여름철에 관측된 장마 시작 지수와 비교하여 예측성능을 분석하는 제 4 단계;

를 포함하는 것을 특징으로 하는 장마 시작의 물리 통계 예측 방법.

Related Patent (2/2)

[청구항 2]

제 1 항에 있어서, **상기 선행인자**는,

한반도 남동쪽에 고기압성 아노말리를 형성하여 고기압성 아노말리의 북서 가장자리의 남서풍에 의해 한반도 남쪽으로 유입되는 수증기량을 증가시켜 장마가 시작하도록 하는 **북적도 중앙태평양 지수**(A) 및 **북반구 태평양 지수**(C)와, 한반도 남쪽의 고기압성 아노말리를 형성하고 서쪽으로 확장하여 장마가 시작하도록 하는 **북대서양지수**(B)를 포함하는 것을 특징으로 하는 장마 시작의 물리 통계 예측 방법.

감사합니다.

5. References

참고 문헌

1. KAIST SESLAB 강의 자료 - http://seslab.kaist.ac.kr/xe2/page_GBex27