2. Decision Tree (의사결정나무)

2.1 Decision Tree Intro

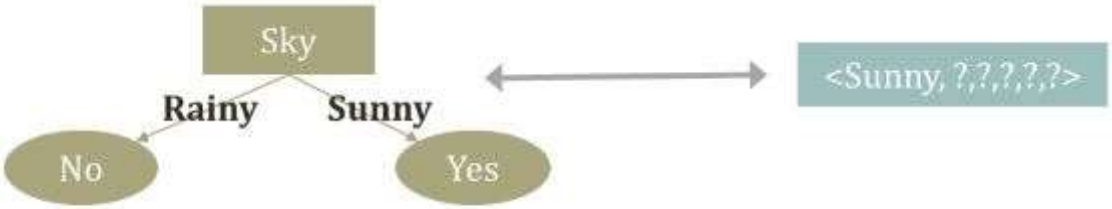


그림 2-10: 의사결정 나무 예시

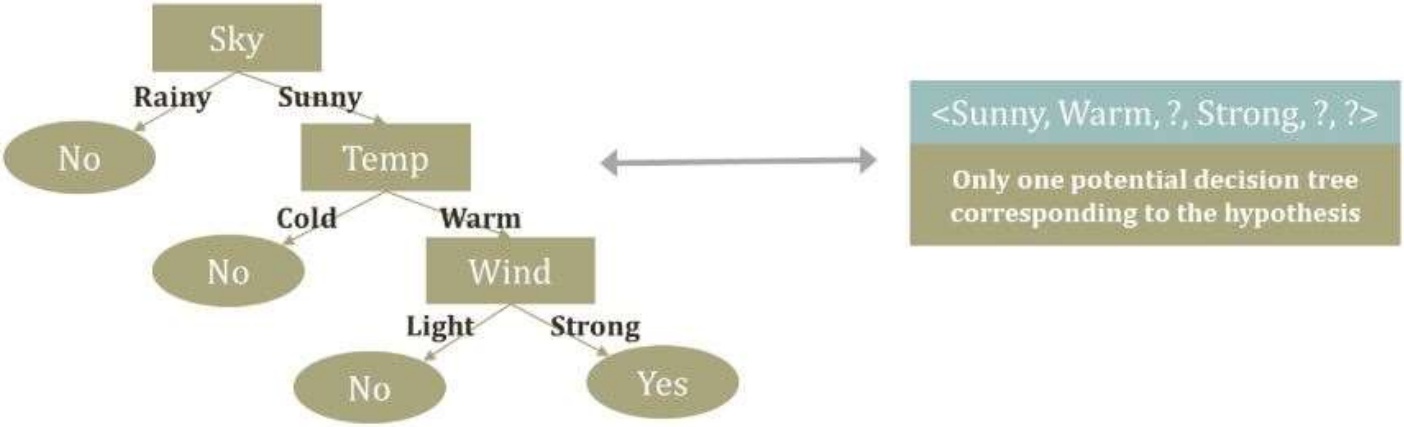


그림 2-11: 의사결정 나무 예시2

- Noise가 포함된 데이터를 통해 목표함수를 찾을 때 규칙기반 알고리즘에 비해 효율적인 방법
- 위는 각각의 Hypothesis를 decision tree로 표현한 것

2.2 Entropy

그림 2-12: UCI 신용평가 데이터에서 A1과 A9 Attribute의 분포

- 어떤 attribute를 더 잘 체크할 수 있는지 알려주는 지표
  - 이 분류 방법은 얼마나 정확하게 분류해줄 수 있는가?
    = 이 분류 방법은 얼마나 불확실성(Uncertainty)을 줄여주는가?

- *가령 위의 예시의 경우, A1 이라는 방식으로 690(307+,383-)를 구별할 경우 attribute (+,-)의 판별을 명확히 하지 못함. A9의 경우 상대적으로 attribute의 판별이 명확함*

- Entropy는 이 불확실성을 측정하는 방법중 하나

- *항상 앞면만 나오는 동전 => 불확실성 X => 매우 낮은 Entropy*
- *양면 모두 1/2의 확률로 나오는 동전 => 불확실성 높음 => 매우 높은 Entropy*

> - *Entropy* $H(X)$ = -$\Sigma_x P(X =$x$) \log_b P(X = x)$
> - *Conditional Entropy* $H(X)$ = -$\Sigma_x P(X =$x$) H(Y \mid X = x)$
> = $\Sigma_x P(X =$x$)\{-\Sigma_Y P(Y = y \mid X = x) \log_b P(Y = y \mid X = x)\}$

- 수식 전개

$$H(X) = -\sum_X P(X = x) \log_b P(X = x)$$

위의 정의로 부터

$$H(Y|X) = -\sum_Y P(Y = y|X) \log_b P(Y = y|X)$$ $

$$\mathrm{H}(Y) = \sum_{y \in \mathcal{Y}} \Pr(Y = y)\,\mathrm{I}y) = -\sum_{y \in \mathcal{Y}} p_Y(y) \log_2 p_Y(y)$$”

위의 정의에 의해서

$$\mathrm{H}(Y|X = x) = -\sum_{y \in \mathcal{Y}} \Pr(Y = y|X = x) \log_2 \Pr(Y = y|X = x)$$

$H(Y|X)$ 는 $H(Y|X = x)$의 엔트로피에서 X 에 있는 모든 x에 대한 엔트로피 값의 평균값을 의미함.

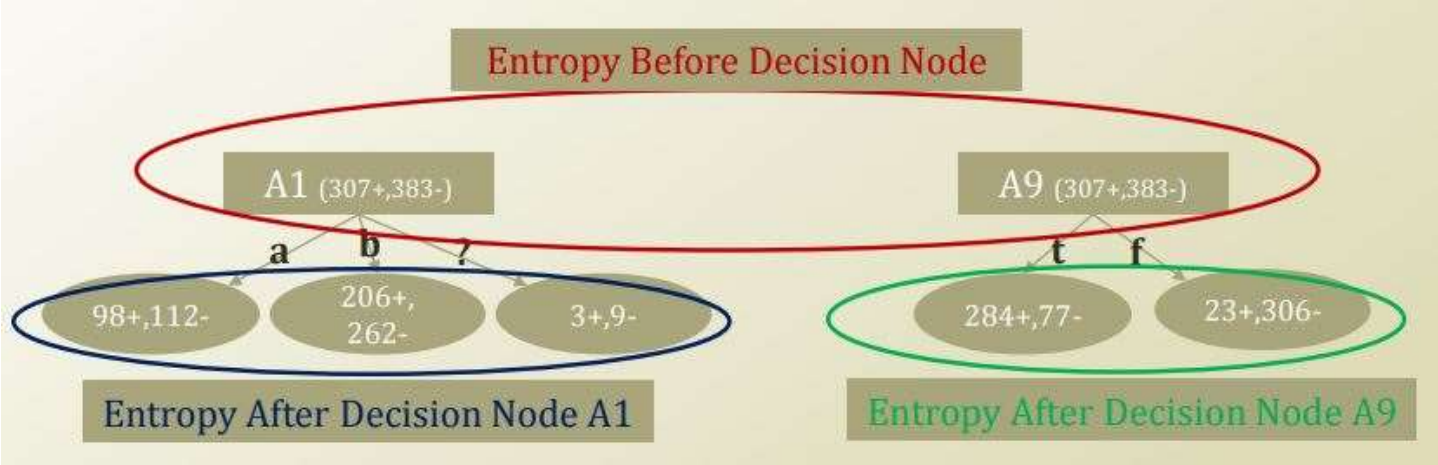
혹은 아래와 같은 수식으로도 이해 할 수 있음(The law of total probability)

$$P(A) = \sum_n P(A \mid B_n) P(B_n)$$

따라서 아래와 같이 수식 전개가 가능함.

$$
\begin{aligned}
\mathrm{H}(Y|X) \ &\equiv \sum_{x \in \mathcal{X}} p(x)\, \mathrm{H}(Y|X=x) \\
&= -\sum_{x \in \mathcal{X}} p(x) \sum_{y \in \mathcal{Y}} p(y|x)\ \log\, p(y|x) \\
&= -\sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x,y)\ \log\, p(y|x) \\
&= -\sum_{x \in \mathcal{X}, y \in \mathcal{Y}} p(x,y) \log\, p(y|x) \\
&= -\sum_{x \in \mathcal{X}, y \in \mathcal{Y}} p(x,y) \log \frac{p(x,y)}{p(x)}. \\
&= \sum_{x \in \mathcal{X}, y \in \mathcal{Y}} p(x,y) \log \frac{p(x)}{p(x,y)}.
\end{aligned}
$$

2.3 Information Gain



- $\mathrm{H(Y)} = -\sum_{Y \in \{+,-\}} P(Y=y) log_2 P(Y=y)$
- $\mathrm{H(Y|A1)} = \sum_{X \in \{a,b,?\}} \sum_{Y \in \{+,-\}} P(A1=x, Y=y) log_2 \frac{P(A1=x)}{P(A1=x,Y=y)}$
- $\mathrm{H(Y|A9)} = \sum_{X \in \{t,f\}} \sum_{Y \in \{+,-\}} P(A9=x, Y=y) log_2 \frac{P(A9=x)}{P(A9=x,Y=y)}$

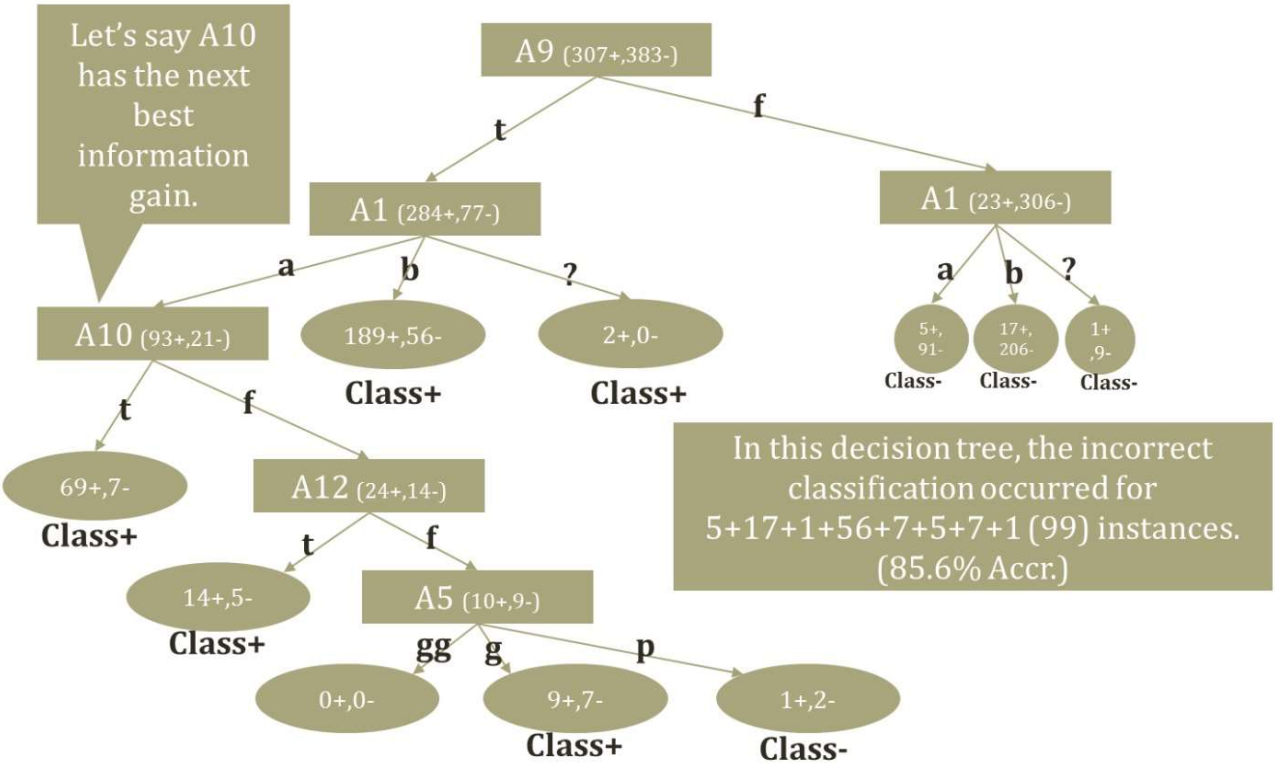
- A1, A9 적용 이전의 Entropy는 $H(Y)$
- A1, A9을 적용한 이후의 Entropy는 $H(Y \mid A1), H(Y \mid A9)$

> Information Gain IG = $H(Y) - H(Y \mid A_i)$
> IG : 특정 attribute에 A_i 라는 condition을 썼을 때의 entropy 차이

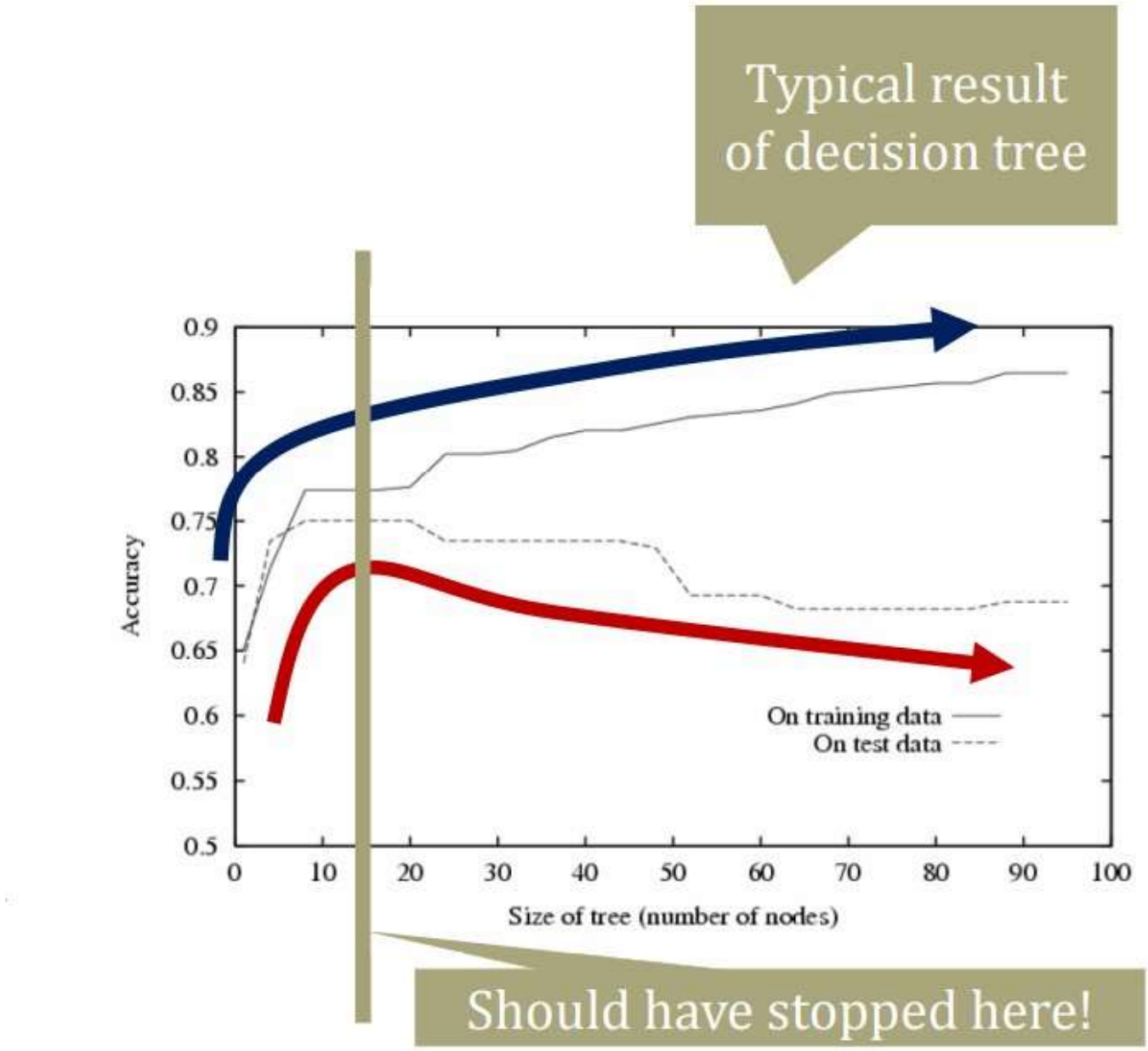
- 위 decision tree에서는 A9 적용 결과 상대적으로 +/-의 분류가 확실해졌기 때문에 IG는 A1에 비해 더 클 것

2.4 Top-Down Induction Algorithm

- ID3, C4.5 CART ...
- ID3 algorithm



- Open Node 생성 (A9)
- 모든 instance를 A9으로 분류
  - Open node 선택
  - 가장 낮은 IG를 갖는 variable 선택
    - variable을 기준으로 instance를 sort
    - sort 결과를 기준으로 instance를 새 branch node로 이동
    - 만약 새 node에 들어간 instance가 모두 동일한 class라면 close node
- 위 과정을 통해 관측한 dataset을 반영한 decision tree 생성 가능



- overfitting 으로 이해하고 있는 현상이 decision tree에서도 발생

3. Linear Regression

3.1 Linear Regression?

- 통계기반의 기계학습의 기초
- hypothesis $h : \hat{f}(x;\theta) = \theta_0 + \Sigma_{i-1}^{n}\theta_i x_i = \Sigma_{i=0}^{n}\theta_i x_i$
- $\Sigma_{i=0}^{n}\theta_i x_i$ 에서 x는 알고 있는 값이므로, θ를 정하는 것이 선형회귀의 목적

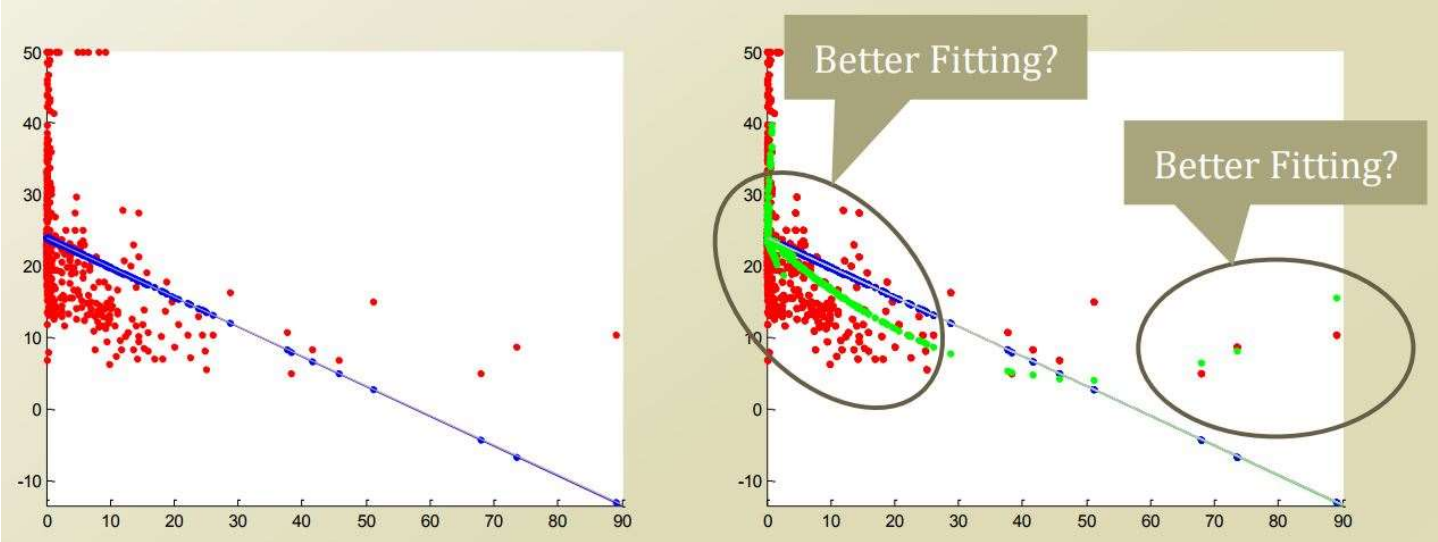
3.2 Finding θ in Linear Regression

- Linear Regression을 통해 좋은 hypothesis를 만들기 위해서는 최적화된 θ를 얻어야 함
  - $\hat{f}(x;\theta) = \Sigma_{i=0}^{n}\theta_i x_i -> \hat{f} = X\theta$

- $X = A_{m,n} = \begin{pmatrix} 1 & \cdots & x_n^1 \\ \vdots & \ddots & \vdots \\ 1 & \cdots & a_n^D \end{pmatrix}, \theta = \begin{pmatrix} \theta_0 \\ \theta_1 \\ \cdots \\ \theta_n \end{pmatrix}$

- hypothesis $\hat{f}$와는 달리 실제 관측된 **f**는 error가 포함되어있음
  - $f(x;\theta) = \Sigma_{i=0}^{n}\theta_i x_i + e = y -> \hat{f} = X\theta + e = Y$
- hypothesis $\hat{f}$와 실제 관측 f의 차이는 noise (error)를 뜻하며, 이를 최소화 하면 최적화된 Linear Regression 결과를 얻을 수 있음
  - $argmin_\theta(f - \hat{f})^2 = argmin_\theta(Y - X\theta)^2 = argmin_\theta(Y - X\theta)^T(Y - X\theta) = argmin_\theta(\theta^T X^T X\theta - 2\theta^T X^T Y + Y^T Y) = argmin_\theta(\theta^T X^T X\theta - 2\theta^T X^T Y)$
  - 이를 만족하는 (= $(f - \hat{f})^2$가 제일 작은 값을 갖도록 하는) $\hat{\theta}$를 구해야 함

- $\hat{\theta}$는 θ에 대해서 미분했을 때 0을 갖는 값(= 극소점)이라고 추론할 수 있음
  - $\hat{\theta} = argmin_\theta(\theta^T X^T X\theta - 2\theta^T X^T Y)$
  - $\nabla_\theta(\theta^T X^T X\theta - 2\theta^T X^T Y) = 0$
  - $\theta = (X^T X)^{-1} X^T Y$



- $\hat{f}(x;\theta) = \Sigma_{i=0}^{n}\theta_i x_i$ 에서 x_i대신에 임의의 기저함수(basis function) $\Phi(x)$ 로 치환하여 Non-Linear한 f로 regression 시킬 수 있음
- 위 그림에서 오른쪽의 녹색으로 표현된 것은 x_i 차수를 높인 결과
- 기저함수로 쓰이는 것은 다음 등이 있으며, 대부분 입력에 따라 일정 값 범위를 벗어나지 않고 진동하는 특징이 있다.(=> wavelets, 소파동)
  - 가우시안 $\Phi(x) = exp\{\frac{(x-\mu_j)^2}{2s^2}\}$
  - 시그모이드 $\Phi(x) = \sigma\{\frac{(x-\mu_j)^2}{s}\}$
    - $\sigma(a) = \frac{1}{1+exp(-a)}$

3.3 Why Use (Linear) Regression

- 최근에 와서는 성능상의 한계로 덜 쓰이고 있음

    - data가 많아질 수록 error가 발생하고, 이를 줄이기 위해서는 모델이 매우 복잡해짐
- 발생할 수 있는 error의 범위가 예측되기 때문에, 이러한 문제가 허용되는 경우에는 쓰임