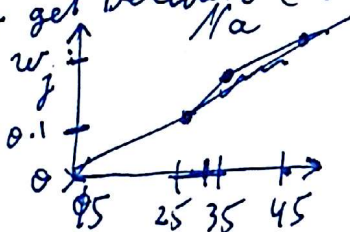


## Criterion

- id: int
- name: str
- min: double
- max: double
- scale: list < double >

## Partial V/Fct

- criterion
- intervals: list < double >
- + get scale(): list < double >
- + get PValue(double x): double
- + get Derivative(double x): double
- + get StartSegment(id): id
- + get Weight(): double



## Utils

- + getScale(min, max: double): list < double >
- + getStartSegment(id): id
- + getWeight(): double

28 →

$$b = 0.1$$

$$a = \frac{0.1}{10} = \frac{0.2 - 0.1}{35 - 25}$$

## Alt

- id: int
- name
- evaluations: map < criterion, double >

## PblGenerator

- nbCriteria: int
- nbAlternatives: int
- random: bool
- criteria: list < criterion >
- alternatives: list < alt >
- + genCriteria(int nb: int, min: double, max: double, nbCriteria: int): list < criterion >
- + genAlternatives(int nb: int, min: double, max: double, nbCriteria: int): list < alt >
- + setCriteria(list < criterion >)
- + setAlternatives(list < alt >)
- + setPref(list < alt >)

## V/Fct

- list < P/VF >
- + get Value(a: Alternative): double

## Solver UTAS

- criteria
- alternatives // ordered
- scales: map < criterion, double >
- + find V(): Value Fct

Gen value Fct: (3 cuts, 4 cuts)

1) generate (3, 1) → 3 weights

2) generate (4,  $w_2$ )  $\rightarrow$  0.4  $\rightarrow$  0.05  
 $\uparrow$   
 nb cuts  $\rightarrow$  0.3  
 $\uparrow$   
 0.05

min Value: 10

max Value: 30 40

3) compute segments:  $\{(10, 0), ($

3) gen scale (10, 40, 4) →  $\{10, 20, 30, 40\}$

4) compute deduce segments:  $\{(10, 0), (20, 0.05), (30, 0.35), (40, 0.40)\}$

5) obtain Partial Value Fct

1641

Val Fct Generator

- criteria // setter

+ gen VFct(): list VFct

Simulation

1) gen DM  $\rightarrow$  VFct  $\rightarrow$   $v^R$

2) obtain  $v^T$   $\rightarrow$  gen alts (with lbl (gen))

3) rank alts with  $v^R$

4) UT 4 stars  $\rightarrow v^T$

5) gen more alts, compare ( $v^R, v^T$ )