

# C4.5 Reference Code 분석

11.24 세미나

201002721 | 안병민

# INDEX

- 개요
- 각 헤더 파일의 역할
- 각 헤더 파일 내부 함수 및 변수 설명

01

## Reference source 구성

02

03

besttree.h

getopt.h

testrules.c

build.h

info.h

trees.h

c4.5.c

makerules.c

getnames.h

classify.h

prune.h

subset.h

confmat.h

prunerule.c

getdata.h

contin.h

rules.c

st-thresh.h

discr.h

siftrules.c

genrules.c

genlogs.c

sort.h

stats.h

01

02

03

<b>c4.5.c</b>	Main routine
<b>info.h</b>	Information과 Information gain 계산 및 출력
<b>sort.h</b>	Sorting utilites
<b>discr.h</b>	Discrete valued attribute에 대한 시험 및 평가
<b>contin.h</b>	Continuous valued attribute에 대한 시험 및 평가
<b>subset.h</b>	Discrete attribute의 subset에 대한 평가
<b>classify.h</b>	의사 결정 트리(decision tree)를 이용하여 case description을 결정

01

02

03

## info.h

### 함수 구성

```
float Worth(float ThisInfo, float ThisGain, float MinGain);
```

```
void ResetFreq(DiscrValue MaxVal);
```

```
void PrintDistribution(Attribute Att, DiscrValue MaxVal, Boolean ShowNames);
```

```
float TotalInfo(ItemCount V[], DiscrValue MinVal, DiscrValue MaxVal);
```

```
float ComputeGain(float BaseInfo, float UnknFrac, DiscrValue MaxVal,  
                  ItemCount TotalItems);
```

```
typedef short   DiscrValue;
```

```
typedef double  ItemCount;
```

```
typedef char    Boolean;
```

```
typedef short   Attribute;
```

```
typedef union   _attribute_value
```

```
{
```

```
    DiscrValue  _discr_val;
```

```
    float       _cont_val;
```

```
}
```

```
    AttValue, *Description;
```

01

02

03

## sort.h

### 함수 구성

```
void Quicksort(ItemNo Fp, ItemNo Lp, Attribute Att, void (*Exchange)())
```

typedef int ItemNo;



01

02

03

## discr.h


### 함수 구성

`void ComputeFrequencies(Attribute Att, ItemNo Fp, ItemNo Lp);`

`void EvalDiscreteAtt(Attribute Att, ItemNo Fp, ItemNo Lp, ItemCount Items);`

`float DiscrKnownBaseInfo(DiscrValue KnownItems, ItemCount MaxVal);`

`void DiscreteTest(Tree Node, Attribute Att);`



```
typedef struct _tree_record *Tree;
typedef struct _tree_record
{
    short NodeType;
    Class NoLeaf;
    ItemCount Items,
                *ClassDist,
                Errors;
    Attribute Tested;

    short Forks;
    float Cut,
           Lower,
           Upper;
    Set *Subset;
    Tree *Branch;
}
TreeRec;
```

01

02

03

## contin.h

### 함수 구성

```
void ContinTest(Tree Node, Attribute Att);
```

```
float GreatestValueBelow(Attribute Att, float t);
```

```
void EvalContinuousAtt(Attribute Att, ItemNo Fp, ItemNo Lp);
```



01

02

03

## subset.h

### 전역 변수

```
ItemCount *Slice1;
          *Slice2;
```

```
short *Subsets;
```

```
Set **Subset;
```

```
typedef char *Set;
```

### 함수 구성

```
void SubsetTest(Tree Node, Attribute Att);
```

```
void PrintSubset(Attribute Att, Set Ss);
```

```
void Uncombine(DiscrValue x, DiscrValue y);
```

```
void Combine(DiscrValue x, DiscrValue y, DiscrValue Last);
```

```
void EvalSubset(Attribute Att, ItemNo Fp, ItemNo Lp, ItemCount Items);
```

01

02

03

## classify.h

전역 변수

```
float*ClassSum = Nil;
```

함수 구성

```
ClassNo Category(Description CaseDesc, Tree DecisionTree);
```

```
void Classify(Description CaseDesc, Tree T, float Weight);
```

11.24 세미나

**Thank you**

**Q & A**