

1 確率的探索による機械学習

表 formula[a][b] 配列の要素と命題の対応関係

a \ b	0	1	2	3	4	5	6	7
0	p1	¬ p1	p2	¬ p2	p3	¬ p3	p4	¬ p4
1	p1	¬ p1	p2	¬ p2	p3	¬ p3	p4	¬ p4
2	p1	¬ p1	p2	¬ p2	p3	¬ p3	p4	¬ p4

表のデータ構造にランダムな値をセットすることで、分類知識を生成し、その結果を評価することを繰り返せば、優秀な分類知識を得ることができます。

このような、知識を生成してその結果を検査することでよりよい知識を得ようとする方法を生成と検査 (generate and test) による分類知識の獲得による方法と呼びます。はじめての機械学習 小高知宏著 117 頁より

2 生成と検査によって分類知識を獲得するためのアルゴリズム

学習データセットの読み込み

乱数による分類知識生成

分類知識の評価

乱数による分類知識生成へ

3 分類知識を機械学習するプログラム

`eml.c`

```
/*
  eml.c
  分類知識を機械学習するプログラム
  入力は標準入力から与え、出力は標準出力に出力します
  使い方
  ./eml <(入力ファイル名)> <(出力ファイル名)>
  入力ファイルでは、属性とカテゴリを 0/1 で記述します
  出力ファイルには、学習結果を表示します
*/
```

```

#include <stdio.h>
#include <stdlib.h>
#include <string.h>

#define TEXTLENGTH 256
#define DATANO 256
#define ATTNO 4
#define TERMNO 3
#define YES 1
#define NO 0
#define SEED 65535
#define REPNO 1000
#define THVAL 0.7

int readdata(int dataset[DATANO][ATTNO+1]);

int rnd();

void genformula(int formula [TERMNO][ATTNO * 2]);

double evalformula(int formula [TERMNO][ATTNO*2],
                   int dataset[DATANO][ATTNO + 1],int no);

void purformula(int formula[TERMNO][ATTNO * 2]);

int evals(int formula[TERMNO][ATTNO*2],int subset[ATTNO+1]);

int notval(int v);

int main(int argc, char *argv[])
{
    int no;
    int i;
    int dataset[DATANO + 1] = {0};
    int formula[TERMNO][ATTNO * 2] = {0};

    double score = 0;
    srand(SEED);

    no = readdata(dataset);

    for(i = 0; i < REPNO; ++i){
        genformula(formula);
        score = evalformula(formula,dataset,no);
        if(score > THVAL){
            printf("\n スコア:%lf\n", score);
            purformula(formula);
        }
    }
}

```

```

    return 0;
}

int readdata(int dataset[DATANO][ATTNO+1])
{
    int no = 0;
    int att = 0;
    int data;
    while(scanf("%d", &data) != EOF){
        dataset[no][att] = data / 10000;
        ++att;
        dataset[no][att] = (data % 10000) / 1000;
        ++att;
        dataset[no][att] = (data % 1000) / 100;
        ++att;
        dataset[no][att] = (data % 100) / 10;
        ++att;
        dataset[no][att] = (data % 10);
        ++no;
        att = 0;
    }

    return no;
}

int rnd()
{
    int rndno;

    while((rndno = ((double) rand() / RAND_MAX) * 2) == 2);

    return rndno;
}

void genformula(int formula [TERMNO][ATTNO * 2])
{
    int i,j;

    for(i = 0; i < TERMNO; ++i){
        for(j = 0; j < ATTNO * 2; ++j){
            formula[i][j] = rnd();
        }
    }
}

void purformula(int formula[TERMNO][ATTNO * 2])
{
    int i,j;

```

```

for(i = 0; i < TERMNO; ++i){
    for(j = 0; j < ATTNO * 2; ++j){
        printf("%d ", formula[i][j]);
    }
    printf("\n");
}
}

double evalformula(int formula[TERMNO][ATTNO*2],
                  int dataset[DATANO][ATTNO+1],int no)
{
    int i;
    int point = 0;

    for(i = 0; i < no; ++i){
        point += evals(formula, dataset[i]);
    }

    return (double)point/(double)no;
}

int evals(int formula[TERMNO][ATTNO*2],int subset[ATTNO+1])
{
    int orval,andval,result;
    int i,j;

    andval = YES;

    for(i = 0; i < TERMNO; ++i){
        orval = NO;

        for(j = 0; j < ATTNO * 2; ++j){
            if((j % 2) == 0){
                orval += subset[j / 2] * formula[i][j];
            }else{
                orval += notval(subset[j / 2]) * formula[i][j];
            }
        }

        if(orval == NO){
            andval = NO;
        }else{
            andval *= YES;
        }
    }

    if(andval == subset[ATTNO]){
        return 1;
    }
}

```

```
    }else{
        return 0;
    }
}

int notval(int v)
{
    if(v == YES){
        return NO;
    }else{
        return YES;
    }
}
```

4 「eml.c」が受け取る学習データ ex.txt

```
11101
10101
01101
11101
00100
11000
11110
11000
11010
00110
```

5 「eml」の操作方法

「eml」の操作方法の例

```
./eml < ex.txt
```

上記の「./」は、Linuxでのプログラムを実行する時の例です。
ウィンドウズでは、「eml < ex.txt Enter」と入力してください。

6 「eml」の実行結果の例

```
スコア:0.900000
0 0 0 0 0 0 1
```

1 0 1 0 0 0 1 1
0 0 0 0 1 0 0 0

スコア:0.800000
0 0 0 0 0 0 0 1
1 0 1 0 0 0 1 0
0 0 1 0 1 1 1 1

スコア:0.800000
0 1 0 0 1 0 0 0
1 1 0 0 1 1 0 0
0 1 0 0 0 1 0 1

スコア:0.800000
0 1 0 1 0 1 0 1
0 1 0 0 0 1 0 1
0 1 0 0 1 0 0 0

スコア:0.800000
0 0 0 0 1 0 1 0
0 1 1 1 1 0 0 1
0 0 0 1 0 0 0 1

スコア:0.800000
1 0 1 0 0 1 1 0
1 0 0 1 0 1 0 1
0 1 0 1 1 0 0 0

スコア:0.900000
0 0 0 0 0 1 0 1
0 0 0 0 1 0 0 1
0 0 0 1 1 0 1 0

スコア:0.800000
0 0 0 0 1 0 1 0
1 1 0 0 1 0 0 1
0 1 0 1 0 0 0 1

スコア:0.800000

0 1 0 0 0 0 0 1
0 0 0 0 1 0 0 0
1 1 1 1 1 0 0 1

スコア:0.800000
0 1 1 0 1 1 1 1
1 0 1 0 0 0 0 0
0 0 0 1 0 0 0 1

スコア:0.900000
0 1 0 0 1 0 0 0
0 0 0 0 0 1 0 1
0 1 1 1 1 1 1 0

スコア:0.800000
0 1 0 1 0 0 0 0
1 0 1 0 0 1 0 0
0 1 1 0 1 0 0 0

スコア:0.800000
0 0 0 1 1 0 1 0
0 1 0 1 0 0 0 1
1 0 0 1 1 1 0 1

スコア:0.900000
0 0 0 1 1 0 1 0
0 0 0 0 0 0 0 1
1 1 1 0 0 0 0 0

スコア:0.800000
0 1 0 0 0 1 0 1
0 0 0 1 1 0 0 0
1 1 1 1 1 1 1 0

スコア:0.800000
0 0 0 1 1 0 0 0
1 1 0 0 0 1 1 1
0 0 0 1 0 0 0 1