

ベイジアンネットワークを利用したプログラミング科目初回テストの解析

占部 弘治 *1

Email: coji@ect.niihama-nct.ac.jp

*1: 新居浜工業高等専門学校電子制御工学科

◎ Key Words ベイジアンネットワーク 初回テスト プログラミング科目

1 はじめに

大学・高等専門学校の電子・情報・制御系の学科において、プログラミングの基礎を習得が重要視されるのは、問題解決のツールとして有用であるからであり、数値計算やシミュレーション、マイコンによる自動制御・計測コンピュータシステムの開発やネットワークの運用など工学のさまざまな分野で利用することが多いからである。また、高等学校の「情報」、中学校の「技術」の科目においてもプログラミングの基礎を学ぶことが学習指導要領に示されており、専門以外の分野においてもプログラミングの基礎を習得することは必要になってくる。そのため、プログラミングの基礎の習得が十分でない学生を早期に発見し、ケアをすることの重要性は高い。

そこで、プログラミングの科目を受講し、発展的な内容の科目になる直前で実施したテストの解析し、習得が十分でない学生が発見できるのではないかと考えその解析にベイジアンネットワーク⁽¹⁾⁽²⁾を用いることにした。ベイジアンネットワークとは複数の確率変数をグラフ構造で表し、個々の変数間の関係を条件付確率で表したモデルである。ネットワークの構造は学習と推論によってデータの関連性から構築され、それによって各変数の関連の強さを明らかにする。

新居浜高専の電子制御工学科では、第3学年の第1回目の講義の際に実施したテスト（初回テスト）を実施し、6年間の結果がある。本報告では、これらのテスト結果をベイジアンネットワークで学習し、ネットワークを構築する。この結果を元に、各問題と到達度についての考察を行い、初回テストによる習熟の遅れのおよび3年次でケアの必要な学生の発見をすることができないかを考える。

2 開講されているプログラミング基礎科目

新居浜高専の電子制御工学科において、プログラミングの基礎を習得する科目には主に2年生で開講されている「情報処理2」、3年生で開講されている「情報処理3」がある。（「情報処理1」は1年生で開講されている科目であり、これはコンピュータのハードウェアとソフトウェアの基礎知識を習得する科目であり、プログラミングについては紹介程度の内容しか実施していない。）

「情報処理2」では、使用言語をCとし、プログラミング環境の利用方法から始まり、選択や反復、関数などのプログラムの制御、配列やポインタ・構造体などの変数やデータの取り扱いの習得を目的としている。「情報処理3」ではC言語の習得を礎に、使用言語をC++に変え、クラスの使用法、オブジェクト指向の概念、Standard Template Library (STL) の利用方法の習得を目標に開講されている。このように「情報処理3」を開始する前に、「情報処理2」におけるプログラミングの学生習得の状況を把握することは重要であると考えた。

3 初回テスト

そこで、「情報処理2」の最終成績とは別にプログラミングの習得状況を把握するために、「情報処理3」の第1回の講義において抜き打ちのテスト（初回テスト）を実施することにした。

この初回テストに用いた問題を図2に示した。それぞれの問題は以下の項目の理解を調べる目的で出題している。

- 反復処理とその中断が理解できているか（問題 1 2）
- ポインタと配列の関係が理解できているか（問題 3）

- 基礎的な反復処理と計算のプログラムが記述できるか（問題 4）
- プログラミングを記述する場合に留意すべき点を知っているか（問題 5）

問題 5 の回答としては、「配列を使用し、ループで処理」「変数は実数でなく、実数で宣言」「意味のある変数名をつける」の 3 つを想定した。

4 集計結果

平成 21 年度から平成 26 年度の 6 年間、この初回テストを実施した。これらの学生のうち、留学生・転科学生で「情報処理 2」を受講していない学生、第 3 学年の途中で休学または退学した学生を除き、有効な結果として、のべ 235 名の学生のデータを得ることができた。

表 1 に、その採点結果を示す。

表 1: 初回テストの採点結果（正解者数）

| 年度 | 人数 | A | B | C | D | E | F | G |
|----|-----|-----|----|----|-----|----|----|---|
| 21 | 37 | 30 | 8 | 18 | 19 | 5 | 16 | 0 |
| 22 | 41 | 35 | 9 | 12 | 34 | 2 | 8 | 1 |
| 23 | 44 | 21 | 6 | 14 | 34 | 2 | 11 | 1 |
| 24 | 38 | 21 | 8 | 10 | 21 | 1 | 6 | 0 |
| 25 | 39 | 30 | 14 | 29 | 16 | 2 | 7 | 1 |
| 26 | 36 | 27 | 9 | 10 | 14 | 5 | 11 | 1 |
| 合計 | 235 | 164 | 54 | 93 | 138 | 17 | 59 | 4 |

表 1 の記号 A ~ G はそれぞれ初回テストの問題を示し、その対応を表 2 に示す。

表 2: 記号

| 記号 | 説明 |
|----|---|
| A | 初回テスト 問題 1 を正解 |
| B | 初回テスト 問題 2 を正解 |
| C | 初回テスト 問題 3 を正解 |
| D | 初回テスト 問題 4 を正解 |
| E | 初回テスト 問題 5 について、 「配列を使用し、ループで処理」と回答 |
| F | 初回テスト 問題 5 について、 「変数は実数でなく、実数で宣言」と回答 |
| G | 初回テスト 問題 5 について、 「意味のある変数名をつける」と回答 |

また、ある問題が正解のとき、他の問題が正解していた確率を表 3 に、ある問題が不正解のとき、他の問題が正解していた確率を表 4 に示す。記号 R は「情報処理 3」の学年末での成績が 100 満点中 80 点以上を示している。

5 解析結果

このデータを用いて、Weka のベイジアンネットワークエディタに読み込み、ベイジアンネットワークの学習を行った。探索アルゴリズムには Hillclimber（山登り法）を用いた。Weka は ニュージーランドのワイカト大学で開発された機械学習ソフトウェアである⁽³⁾。

この学習によって得られたベイジアンネットワークを図 1 に示す。

ノード G はネットワークに加えられなかった。これは G の正解数が少ないためと考えられる。ネットワークは $R \rightarrow A \rightarrow E$ 、 $R \rightarrow B \rightarrow C$ が形成され、類似問題 A, B 間の関連が小さく、A と E、B と C の間の関連が大きいことが示された。B および C は正解者が少なく、難易度が高い問題である。難易度の高い問題に正解できる学生は他の難易度の高い問題にも正解できるからではないかと考える。A と E の関連の高さは、E が正解できる学生は A も十分理解できて正解できていると考えられる。これらのことから、問題の関連性よりも個々の学生のプログラミング習熟度の違いが影響していると予想する。

6 おわりに

新居浜高専電子制御工学科のプログラミング基礎科目を 1 年間履修した学生を対象に実施したテストの採点結果をベイジアンネットワークの学習によって、その各問題間の関連を調べ、その結果の考察を行った。その考察より問題の類似性よりも学生の個々の習熟の影響が強いのではないかという予想が得られた。

この予想を確かにするために、学生のプログラミングに対する日頃の取組や意欲などの実感に関する調査を行い、解析する必要がある。今後は初回テストと同時に採取しているプログラミングに対する好悪や意欲、授業に対する態度や理解の実感のアンケートを元に解析を進めていく。

参考文献

- (1) Pearl J.: “Probabilistic reasoning in intelligent systems: networks of plausible inference”, California, 1988.
- (2) 本村陽一・岩崎弘利, 「ベイジアンネットワーク技術」, 東京電機大学出版局, 2006.
- (3) Weka 3: Data Mining Software in Java, <http://www.cs.waikato.ac.nz/ml/weka/>

表 3: 問題が正解のとき、他の問題も正解した確率

| | | 正解した確率 | | | | | | | |
|----------------------------|---|--------|-------|-------|-------|-------|-------|-------|-------|
| | | A | B | C | D | E | F | G | R |
| 正 解 し た 問 題 | A | — | 0.280 | 0.482 | 0.659 | 0.104 | 0.293 | 0.024 | 0.640 |
| | B | 0.852 | — | 0.611 | 0.574 | 0.148 | 0.315 | 0.037 | 0.630 |
| | C | 0.849 | 0.355 | — | 0.710 | 0.140 | 0.398 | 0.032 | 0.753 |
| | D | 0.783 | 0.225 | 0.478 | — | 0.094 | 0.290 | 0.029 | 0.710 |
| | E | 1.000 | 0.471 | 0.765 | 0.765 | — | 0.412 | 0.059 | 0.765 |
| | F | 0.814 | 0.288 | 0.627 | 0.678 | 0.119 | — | 0.000 | 0.847 |
| | G | 1.000 | 0.500 | 0.750 | 1.000 | 0.250 | 0.000 | — | 0.750 |
| | R | 0.802 | 0.260 | 0.534 | 0.748 | 0.099 | 0.382 | 0.023 | — |

表 4: 問題が不正解でも、他の問題も正解した確率

| | | 正解した確率 | | | | | | | |
|---------------------------------|---|--------|-------|-------|-------|-------|-------|-------|-------|
| | | A | B | C | D | E | F | G | R |
| 不 正 解 し た 問 題 | A | — | 0.113 | 0.197 | 0.423 | 0.000 | 0.155 | 0.000 | 0.366 |
| | B | 0.652 | — | 0.331 | 0.591 | 0.050 | 0.232 | 0.011 | 0.536 |
| | C | 0.599 | 0.148 | — | 0.507 | 0.028 | 0.155 | 0.007 | 0.430 |
| | D | 0.577 | 0.237 | 0.278 | — | 0.041 | 0.196 | 0.000 | 0.340 |
| | E | 0.674 | 0.211 | 0.367 | 0.573 | — | 0.239 | 0.014 | 0.541 |
| | F | 0.659 | 0.210 | 0.318 | 0.557 | 0.057 | — | 0.023 | 0.460 |
| | G | 0.693 | 0.225 | 0.390 | 0.580 | 0.069 | 0.255 | — | 0.554 |
| | R | 0.567 | 0.192 | 0.221 | 0.385 | 0.038 | 0.087 | 0.010 | — |

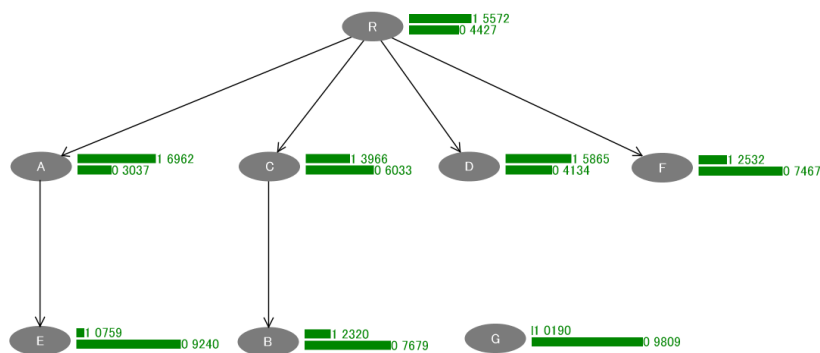


図 1: 学習されたベイジアンネットワーク

- 1 次のプログラムを実行したときの出力結果を示してください。

```
#include <stdio.h>
int main(void) {
    int a, i;
    a = 5;
    for (i = 0; i < 9; i++) {
        if (i > a) {
            break;
        }
    }
    printf("%d\n", i);
}
```

- 2 次のプログラムを実行したときの出力結果を示してください。

```
#include <stdio.h>
int main(void) {
    int a, i;
    a = 15;
    for (i = 0; i < 9; i++) {
        if (i > a) {
            break;
        }
    }
    printf("%d\n", i);
}
```

- 3 次のプログラムを実行したときの出力結果を示してください。

```
#include <stdio.h>
int main(void) {
    int *a;
    int b[] = {1, 2, 3, 4, 5};
    a = b;
    printf("%d\n", *(a+3));
    return 0;
}
```

- 4 任意の整数をキーボードから入力し、1 から入力された整数までの整数の総和を表示するプログラムを作りなさい。

- 5 次のプログラムは3つの整数を入力し、その合計を求め、それぞれの整数が全体の何%であるかを計算するプログラムです。このプログラムの問題点は何か書きなさい。(3つ)

```
#include <stdio.h>

int main(void) {

    int a1, a2, a3;
    int b;
    int c1, c2, c3;

    printf("a1=?");
    scanf("%d", &a1);
    printf("a2=?");
    scanf("%d", &a2);
    printf("a3=?");
    scanf("%d", &a3);

    b = a1 + a2 + a3;

    printf("合計 %d\n", b);

    c1 = 100 * a1 / b;
    c2 = 100 * a2 / b;
    c3 = 100 * a3 / b;

    printf("a1 の割合 %d\n", c1);
    printf("a2 の割合 %d\n", c2);
    printf("a3 の割合 %d\n", c3);

    return 0;
}
```

図 2: 初回テストの問題