計算科学[1]後期期末試験

龍谷大学理工学部数理情報学科 2003年01月22日樋口さぶろお[2]

- **理由, 過程を記すように指定がある問以外は**理由, 過程を省略してよいが, どの問も, 理由, 過程が記してある場合, 部分点を与えることがある.
- 問には, 問題に現れる記号だけを使って答えよう. 自分で導入した記号を用いる場合は, 定義を記そう.

1

この問いでは, **すべて理由, 過程を記そう.** 実数に値をとる確率変数 X は, 確率密度

$$p(x) = \begin{cases} 0 & (x < 0) \\ a\mathrm{e}^{-ax} & (x \geq 0) \end{cases} \tag{1}$$

にしたがう. ただし $a > 0$ は定数.

1. $0 \leq x < +1$ の範囲の x が得られる確率を求めよう.
2. X の平均値 $E(X)$ を求めよう.
3. X の分散 $V(X)$ を求めよう.

2

1次元的に無限に広がった土地を考える. 点 $x = \ldots, -3, -2, -1, 0, +1, +2, +3, \ldots$ には0個体または1個体の生物が生息することができる.

各時刻 $t = 0, 1, 2, \ldots$ に, 生物の生息状況は, 次のルールにしたがって変化する.

現在の時刻 t での, 地点 x の個体数	現在の時刻 t での, 両隣の地点 $x \pm 1$ の個体数の合計	次の時刻 $t+1$ での 地点 x の個体数
0	0,2	0
0	1	1
1	0,2	1
1	1	0

時刻 t での 地点 x の個体数を $n(x, t)$ と表わす. 最初の時刻 $t = 0$ には

$$n(x, 0) = \begin{cases} 1 & (x = 0) \\ 0 & (\text{それ以外}) \end{cases} \tag{2}$$

だったとする.

---

[1] http://sparrow.math.ryukoku.ac.jp/~hig/compsci/

[2] mailto:hig@math.ryukoku.ac.jp, http://www.math.ryukoku.ac.jp/~hig/, へや 1-508, でんわ 077-543-7501

1. 時刻 $t = 1, 2, 3, 4$ での生息状況 $n(x, t)$ を, 横が x, 縦が t の表の形で答えよう.

2. 上の繁殖ルールは, 整数 a, b, c を適当に選ぶと,

$$n(x, t+1) = [\{(a \cdot n(x-1, t) + b \cdot n(x, t) + c \cdot n(x+1, t)\} \text{ を 2 で割った余り}] \quad (3)$$

   の形に書き表せる. 適当な a, b, c は多数あるが, 1 組を答えよう.

3

当り外れ法, ランダムサンプリング法の 2 つのモンテカルロ数値積分の方法を講義で学んだ. どちらかの一方 (だけ) のアルゴリズムを, 日本語で説明しよう. プログラムや疑似コードを利用してもよい. 単純な場合 (たとえば 1 変数の場合) に限ってもよい. なお, 2 つの方法の名前の取り違えは減点しない.

4

N 人のランダムウォーカーが, (x, y) 平面を移動している. 座標は整数値に限られ, 範囲は $x = 0, 1, 2, \ldots, A-1$, $y = 0, 1, 2, \ldots, B-1$ である.

このような状況を表現するデータ構造として,

- `int x[N], y[N];` (ラグランジュ表示)
- `int n[A][B];` (オイラー表示)

の 2 つを考えよう.

1. それぞれの表示で, `x[k]`, `y[k]`, `n[i][j]` に格納されている整数は何を表わすか, 日本語で答えよう.

2. $N = 4, A = 4, B = 4$ とする. ランダムウォーカーのうち

   - 1 人が $(x, y) = (1, 2)$ に,
   - 1 人が $(x, y) = (2, 3)$ に,
   - 2 人が $(x, y) = (3, 1)$ に

   いる. この状況を, ラグランジュ表示, オイラー表示でそれぞれ表示したとき配列 `x[]`, `y[]`, `n[][]` に格納されている整数値を, 表の形で答えよう.

3. ラグランジュ表示, オイラー表示の特徴, 利点と欠点を日本語で説明しよう. 次のようなフレーズを使ってもいいかもしれない.

   場所指向, 人指向, ヒーローキャラ, 敵雑魚キャラ, 2 人のウォーカーが衝突したとき, 1 人のウォーカーに着目して追跡するとき,

5

文字列 “エルモ”, “クッキーモンスター”, “ビッグバード” を, それぞれ 1/3 の確率で出力することを 10 回繰り返す (10 回全部同じ名前が表示されることはまれにしかない), 下のようなプログラムを考える.

ただし, `void srand(unsigned int)` は乱数のシード (種) を設定する関数, `int rand(void)` は, 0 以上 `RAND_MAX` 以下の整数値の乱数を返す関数だった.

プログラム例

```
/** 乱数でマペットの名前を選んで出力する */
#include <stdio.h>
#include <stdlib.h>

/** [0,1) 一様乱数を返す関数 */
double get_uniform_random(void){
    return ((double)rand())/((double)RAND_MAX+1.0);
}


int main(int argc, char **argv){
    int kaisu=10;
    int i;
    double r;

    /* 次の値には意味はない. 別の数でも影響を受けない答えを書く */
    unsigned int seed=129078;

    /************** 埋めてね ***********************/

    return 0;
}
```

1. 埋めてね部分に書く内容として, 適切なもの**すべて**を記号で答えよう.

選択肢

(a)

```
srand(seed);
for(i=0; i < kaisu; i++){
    if( get_uniform_random() < 1.0/3.0 ){
        printf("エルモ\n");
    } else if( get_uniform_random() < 2.0/3.0 ){
        printf("クッキーモンスター\n");
    } else {
        printf("ビッグバード\n");
    }
}
```

選択肢

(b)
```
srand(seed);
for(i=0; i < kaisu; i++){
    r=get_uniform_random();
    if( r < 1.0/3.0 ){
         printf("エルモ\n");
    } else if( r < 2.0/3.0 ){
         printf("クッキーモンスター\n");
    } else {
         printf("ビッグバード\n");
    }
}
```

選択肢

(c)
```
for(i=0; i < kaisu; i++){
    srand(seed);
    r=get_uniform_random();
    if( r < 1.0/3.0 ){
         printf("エルモ\n");
    } else if( r < 2.0/3.0 ){
         printf("クッキーモンスター\n");
    } else {
         printf("ビッグバード\n");
    }
}
```

選択肢

(d)
```
for(i=0; i < kaisu; i++){
    srand(seed);
    if( get_uniform_random() < 1.0/3.0 ){
        printf("エルモ\n");
    } else if( get_uniform_random() < 2.0/3.0 ){
        printf("クッキーモンスター\n");
    } else {
        printf("ビッグバード\n");
    }
}
```

選択肢

(e)
```
srand(seed);
r=get_uniform_random();
for(i=0; i < kaisu; i++){
    if( r < 1.0/3.0 ){
         printf("エルモ\n");
    } else if( r < 2.0/3.0 ){
         printf("クッキーモンスター\n");
    } else {
         printf("ビッグバード\n");
    }
}
```

2. 上で選ばなかった選択肢のうちひとつを選んで (選んだ誤答の記号を明示しよう), それがなぜ正しくないか, どのような出力をするか, を日本語で説明しよう.

計算科学[3]後期期末試験略解

龍谷大学理工学部数理情報学科 2003 年 01 月 22 日樋口さぶろお[4]

1

1. 確率は,

$$P(0 \le X < +1) = \int_0^1 a\mathrm{e}^{-ax}\,\mathrm{d}x = 1 - \mathrm{e}^{-a}. \tag{4}$$

2. $E(X) = a^{-1}$. quiz 27 の解答参照.

3. 分散の公式を利用すると便利.

$$\begin{aligned} V(X) =& E((X - E(X))^2) = (\text{公式}) = E(X^2) - E(X)^2 \\ &= \int_0^\infty x^2 a\mathrm{e}^{-ax}\,\mathrm{d}x - (a^{-1})^2 = (\text{部分積分}) = 2a^{-2} - a^{-2} = a^{-2}. \end{aligned} \tag{5}$$

2

1. 下の表で, 範囲の外はすべて 0.

$t\backslash x$	-5	-4	-3	-2	-1	0	$+1$	$+2$	$+3$	$+4$	$+5$
0	0	0	0	0	0	1	0	0	0	0	0
1	0	0	0	0	1	1	1	0	0	0	0
2	0	0	0	1	0	1	0	1	0	0	0
3	0	0	1	1	0	1	0	1	1	0	0
4	0	1	0	0	0	1	0	0	0	1	0

2. この例では, 自分と両隣の個体数の合計が偶数なら 0, 奇数なら 1 となっている. したがって, a, b, c すべてが奇数ならよい. $a = b = c = 1$ は一つの例.

3

定積分 $I = \int_a^b f(x)\mathrm{d}x$ を考える. 簡単のため, $a < b$, $[a, b)$ で $f(x) \ge 0$ とする.

ランダムサンプリング法 $[a, b)$ 一様乱数を十分大きな回数 N だけ得て, $x_1, \cdots, x_N$ とする. 平均 $(b-a)\times\frac{1}{N}\sum_{i=1}^N f(x_i)$ を I の値とする.

当り外れ法 実数 $M > 0$ を, $[a, b)$ で $0 \le f(x) \le M$ であるようにとる. N を十分大きな回数として, $[a, b)$ 一様乱数を N 回得て $x_1, \ldots x_N$ とする. $[0, M)$ 一様乱数を N 回得て $y_1, \ldots y_N$ とする. $i = 1, \ldots, N$ のうち, $y_i \le f(x_i)$ となる i の個数を n とする. $M(b-a)\cdot\frac{n}{N}$ を I の値とする.

[3] `http://sparrow.math.ryukoku.ac.jp/~hig/compsci/`

[4] `mailto:hig@math.ryukoku.ac.jp`, `http://www.math.ryukoku.ac.jp/~hig/`, へや 1-508, でんわ 077-543-7501

4

1. `x[k]`,`y[k]` は k 人目のウォーカーの それぞれ x, y 座標. `n[i][j]` は, 点 $(x, y) = (i, j)$ にいるウォーカーの人数.

2. `x[i]`,`y[i]` は, あわせて順序をかえてもよい.

i	0	1	2	3
x[i] =	1	2	3	3
y[i] =	2	3	1	1

n[i][j] =

i\j	0	1	2	3
0	0	0	0	0
1	0	0	1	0
2	0	0	0	1
3	0	2	0	0

(6)

3. ラグランジュ表示は, 各ウォーカーの位置を記録する人指向の方式. ゲームではヒーローキャラのような, 個性のある少人数を扱うのに便利. しかし, ウォーカー同志の衝突の判定はめんどう.

   オイラー表示は, 各地点にいるウォーカーの人数を記録する場所指向の方式. ゲームでは, 区別のない大勢の敵雑魚キャラの分布を扱うのに便利. しかし, 1 人のウォーカーを区別して行動を追跡することはできない.

5

1. (b) のみ.

2. (a) 最初の `if` 文と次の `if` の条件部に現れる `get_uniform_random()` は別々の乱数を返してしまうので, エルモは $\frac{1}{3}$ の確率で, クッキーモンスターは $(1-\frac{1}{3})\cdot\frac{2}{3} = \frac{4}{9}$ の確率で, ビッグバードは $1-\frac{1}{3}-\frac{4}{9} = \frac{2}{9}$ の確率で出力される.

   (b) 正しい.

   (c) `for` ループの先頭の `srand` で毎回同じシードがセットされるので, 直後の `get_uniform_random()` は毎回同じ実数を返す. したがって, $\frac{1}{3}$ の確率で選ばれた 1 人のマペットが, 10 回繰り返して出力されてしまう.

   (d) (a) と (c) の誤りを同時に犯している. $\frac{1}{3}$ の確率でエルモが 10 回繰り返し出力される. $\frac{4}{9}$ の確率でクッキーモンスターが 10 回繰り返し出力される. $\frac{2}{9}$ の確率でビッグバードが 10 回繰り返し出力される.

   (e) 最初に `get_uniform_random()` が返した値が `r` に保存され 10 回使われるので, $\frac{1}{3}$ の確率で選ばれた 1 人のマペットが, 10 回繰り返して出力されてしまう.