連分数公式を予測するラマヌジャン・マシンの紹介

鈴木 治郎 *1

*1 信州大学 全学教育機構

*1 szkjiro@shinshu-u.ac.jp

キーワード　機械学習，連分数，ラマヌジャン

1　はじめに

数学領域において，情報処理技術による自動で未知の公式の作成を試みることはこれまでも試み続けられており，よく知られたものには，たとえば円周率計算を 16 進数で任意の桁から計算できる BBP 公式がある [2].

$$\pi = \sum_{k=0}^{\infty} \frac{1}{16^k} \left(\frac{4}{8k+1} - \frac{2}{8k+4} - \frac{1}{8k+5} - \frac{1}{8k+6} \right)$$

2020 年に Raayoni らが発表した Ramanujan Machine[1] は，深層学習技術を用いて，数学定数の計算式を与えるものであり，たとえば以下の例は既知の計算式を再発見したものである.

$$\frac{e}{e-2} = 4 - \cfrac{1}{5 - \cfrac{2}{6 - \cfrac{3}{7 - \cfrac{4}{8 - \cdots}}}}$$

以下の例はまだ証明は得られていないものである.

$$\frac{8}{7\zeta(3)} = 1 \cdot 1 - \cfrac{1^6}{3 \cdot 7 - \cfrac{2^6}{5 \cdot 19 - \cfrac{3^6}{7 \cdot 37 - \cdots}}}$$

ただし ζ はゼータ関数を表す．おおまかに言えば，これらの公式も BBP 公式の発見と同様，左辺と右辺の近似値をうまくマッチングさせる点では変わりない．以下ではこの公式生成技術の概要を紹介する.

2　連分数展開

たとえば円周率 π は

$$\pi = 3 + \cfrac{1}{7 + \cfrac{1}{15 + \cdots}}$$

という連分数展開をもち，途中までの展開を分数に直せば

$$3,\ \frac{22}{7},\ \frac{355}{113}, \ldots$$

を得る．連分数展開は，このような実数の近似値を与える方法としても有効である.

連分数を一般に

$$a_0 + \cfrac{a_1}{b_1 + \cfrac{a_2}{b_2} + \cdots}$$

と表すとき， $b_n = 1$ とする連分数を正則連分数という．正則連分数を求めるアルゴリズムは

(1) 整数部 a_n を求める

(2) 小数部の逆数を求めて，(1) から繰り返す

である.

正則連分数に関しては

- 有理数ならば展開は有限である
- 二次無理数ならば展開は周期的である

などがわかる．正則連分数でなければ，円周率においても興味深い関係式がある．たとえば次は Brouncker が与えたものであり，超幾何級数により証明された.

$$\frac{\pi}{4} = 1 + \cfrac{1^2}{2 + \cfrac{3^2}{2 + \cfrac{5^2}{2 + \cdots}}}$$

2.1　連分数展開と漸化式

連分数展開を n 番目までで打ち切って得られる（近似）分数を

$$\frac{p_n}{q_n}$$

で表すとき，

$$p_0 = a_0,\ p_{-1} = 1,\ q_0 = 1,\ q_{-1} = 0$$

とおいて，次の漸化式で計算できる．

$$\begin{cases} p_{n+1} = a_n p_n + b_n p_{n-1} \\ q_{n+1} = a_n q_n + b_n q_{n-1} \end{cases}$$

この漸化式は，行列の形で表せば次の通りである．

$$\begin{pmatrix} p_{n+1} & q_{n+1} \\ p_n & q_n \end{pmatrix} = \begin{pmatrix} a_n & b_n \\ 1 & 0 \end{pmatrix} \begin{pmatrix} p_n & q_n \\ p_{n-1} & q_{n-1} \end{pmatrix}$$

3　ラマヌジャン・マシンの仕組み

3.1　中間一致手続き

数学的定数などを表す c に対する有理式（話を簡単にするために分母・分子とも一次式としたが，論文では一般の多項式を扱っている）

$$\frac{\alpha c + \beta}{\gamma c + \delta}, \quad \alpha, \beta, \gamma, \delta \in \mathbb{Z} \tag{A}$$

のリストおよび適当な $\{a_n\}, \{b_n\}$ に対する近似分数

$$\frac{p_n}{q_n} \tag{B(n)}$$

を考える．このとき，式 (A) と (B(n)) の値の近い組を見つけ出し，さらに計算精度を上げることで計算公式

$$\frac{\alpha c + \beta}{\gamma c + \delta} = \lim_{n \to \infty} \frac{p_n}{q_n}$$

が実際に成り立つ可能性を高くする（中間一致（MTTF; Meet-To-The-Middle）手続き）．

先の Brouncker の公式の場合

$$a_n = n^2,\ b_n = 2$$

である．

3.2　勾配降下手続き

式 (B(n)) の候補を効率よく生成するために機械学習で使われる勾配降下法（GD; Gradient Descent）を適用する．

4　ラマヌジャン・マシンの現在

ラマヌジャン・マシンは 2019 年にオープンソースプロジェクトとして稼働した．

`www.RamanujanMachine.com`

このサーバ上では

- メルセンヌ素数探索プロジェクト GIMPS と同様の計算時間提供ユーザの管理
- 発見された計算公式候補を公開し，証明を募集
- アルゴリズムの改善意見の募集

を行なっており，数学的成果は arXiv を通じて継続的に更新されている．

参考文献

[1] Gal Raayoni et al., The Ramanujan Machine: Automatically Generated Conjectures on Fundamental Constants, https://arxiv.org/abs/1907.00205, 2020.

[2] D.H.Bailey, P.B.Borwein, S.Plouffe, On the Rapid Computation of Various Polylogarithmic Constants, Math. Comp. 66, no.218, 903–913, 1997.