11章　進化する知性

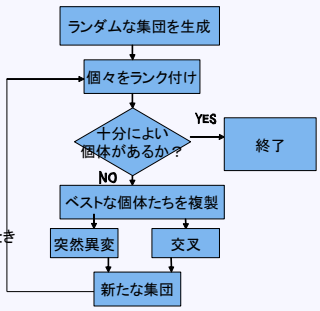
サ　ミンソン

11．1遺伝的プログラミングとは？

- 遺伝的プログラミング　　・プロセス

ー生命の進化に
インスパイアされた
機械学習の技術

・終了条件
-完全な解決法が発見されたとき
-十分によい解決法が発見されたとき
-何世代かにわたり
改善が見られなかったとき
-世代の数が特定の上限値に達したとき

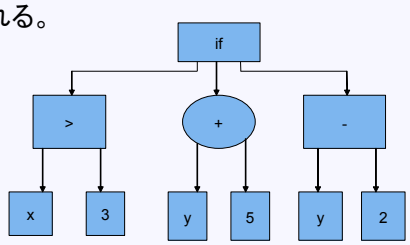


11．1．2遺伝的プログラミング　VS
遺伝アルゴリズム

- 遺伝アルゴリズム：進化的なプレッシャーを利用して最適な解を求める最適化のテクニック
  ーアルゴリズムが尺度基準を選択したら、最適なパラメータを探し出そうとする
- 遺伝的プログラミング：解決法がどの程度よいものかを計る方法を必要とする
  ー解決法はパラメータ集合ではなく、アルゴリズム自身であり、そのすべてのパラメータは進化的なプレッシャーにより自動的に設計される

11．2ツリー構造のプログラム

- ほとんどのプログラミング言語では、インタープリタで読まれる際に、まず解析木の形にされる。



続き

```
def func(x,y);
    if x>3;
        return y + 5;
    else:
        return y - 2;
```

・ツリーは例の解析木よりも非常に複雑な関数にもなりうる。
・ツリー中の上位のノードを参照することでツリーを再帰的にできる

11．2．1Pythonでツリーを表現する

・下の4つのクラスを作成する

ーfwrapper：関数ノードで利用される関数のラッパー
ーnode：子を持っているノード
ーparamnode：プログラムに渡されたパラメータたちの一つを返すだけのノードのクラス
ーconstnode：定数を返すノード

11．2．2ツリーの構築と評価

```
def exampletree():
  return node(ifw,[
   node(gtw,[paramnode(0),constnode(3)]),
   node(addw,[paramnode(1),constnode(5)]),
   node(subw,[paramnode(1),constnode(2)]),
                ]
            )
```

プログラム実行結果

```
>>> import gp
>>> exampletree=gp.exampletree()
>>> exampletree.evaluate([2,3])
1
>>> exampletree.evaluate([5,3])
8
```

11．2．3プログラムを表示する

- 遺伝的プログラミングはプログラムツリーを自動的に作っていくため、その構造がわからない　→　nodeクラスはすべてのノードが関数の名前の文字列を持つよう設計されている。

→表示のための関数はこの文字列と子ノードの表示用の文字列を返せばよい

プログラム例

```
>>> import gp
>>> exampletree=gp.exampletree()
>>> exampletree.display()
if
 isgreater
    p0       // =x
    3
 add
    p1       // =y
    5
 subtract
    p1       // =y
    2
```

11．3最初の集団を作る

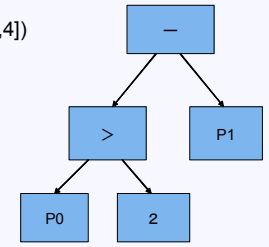
- ランダムなプログラムの集合で作る

ー＞初期の集団により多くの多様性が産まれる

・ランダムなプログラムを作るということは、
ランダムな関数に関連付けられたルートノードを作り、
必要な数のランダムな子ノードを作ること
（子ノードはさらに自身のランダムな子ノードを持つ）

プログラム例

```
>>> reload(gp)
>>> random1=gp.makerandomtree(2)
>>> random1.evaluate([7,1])
0
>>> random1.evaluate([2,4])
-4
>>> random1.display()
subtract
 isgreater
  p0
  2
 p1
```



プログラム例（続き）

```
>>> random2=gp.makerandomtree(2)
>>> random2.evaluate([5,3])
120
>>> random2.evaluate([5,20])
800
>>> random2.display()
if
 p1
 multiply
  p1
  multiply
   p0
   add
    3
    p0
 multiply
  9
  p0
```