オミクスデータから形質発現をシミュレートする
ニューラルネットワークの設計手法の研究

谷口 知世

2021 年 5 月 26 日

目次

第 1 章

序論

test

1.1　研究の背景

1.2　主題

1.3　本稿の構成

第 2 章

手法

2.1 ネットワーク設計に使用したデータ

本研究では、ニューラルネットワーク設計のために複数のデータベース上の遺伝子集合を使用した。そこで本節では、ネットワーク設計に使用したデータベースと、取得したデータについて述べる。

2.1.1 Gene Ontology

Gene Ontology(GO) は、生物学的概念を示す用語 (GO term) と用語間の関係が有向非巡回グラフ (DAG) によって構造化されたデータベースである。DAG の頂点は各 GO term によって構成され、頂点間のエッジは用語間の関係を表している。より概括的な用語である親用語から、より詳細な用語である子用語に向かってエッジが引かれており、ルート以外のどの GO term も複数の親用語、または複数の子用語を持つことができる。各 GO term は、生物学的プロセス (biological process; BP)、細胞の構成要素 (cellular component; CC)、分子機能 (molecular function; MF) のいずれかのカテゴリーに属しており、7 桁の識別子からなる固有の ID が割り振られている。それぞれのカテゴリーは、biological_process(GO:0008150)、cellular_component(GO:0005575)、molecular_function(GO:0003674) という GO 用語をルートとして独立した DAG 構造をとっている。また、各 GO term には関連する遺伝子集合が付加情報として与えられており、遺伝子がどのような生物学的プロセスに関わっているのか、細胞内のどこに影響しているのか、分子レベルでどのように機能しているのかといった情報が示されている。これらの付加情報は、DAG 構造に従って子用語から親用語へと伝播する。したがって、ある

GO term に関連する遺伝子集合は、その先祖の用語全てに関連することとなる。
本実験では、BP、CC、MF を独立のデータベースとして扱う。 BP、CC、MF それぞれについて、GO term ごとに関連する遺伝子集合を抽出する。そして、のネットワークモジュール作成方法に従い、ネットワークモジュールを作成する。

2.1.2 Reactome

2.1.3 Molecular Signatures Database

2.2 ニューラルネットワークの構造

2.2.1 input layer

入力層は、ヒトの各遺伝子 1 つ 1 つに対応するノード $g_m(1 \leq n \leq M)$ で構成され、それぞれのノードには遺伝子発現データが入力される。各サンプルの入力データは $x = \{x_1, x_2, \cdots, x_M\}$ と表せる。

2.2.2 hidden layers

hidden layers は、データベースごとに、遺伝子集合とその親子関係に基づいて作成されたネットワークモジュール $D_k(1 \leq k \leq K)$ で構成される。

書き途中

Gene Ontology や Reactome は、複数の遺伝子集合が DAG 構造に整理されたデータベースであり、遺伝子集合に加え、遺伝子集合の親子関係を取得することができる。このとき、データベース上の遺伝子集合を $S_n(1 \leq n \leq N)$、遺伝子集合 S_n の子遺伝子集合のインデックスの集合を $C_n(1 \leq n \leq N)$ とする。

MSigDB は、複数の遺伝子集合が格納されたデータベースであり、Gene Ontology や Reactome にあるような遺伝子集合間の親子関係は定義されていない。そこで、データベース上の遺伝子集合間の親子関係を作成することとする。データベース上の遺伝子集合を $S_n(1 \leq n \leq N)$ とし、S_n の集合を S とする。このとき、$S_n \in S$、$S_{n'} \in S$ について、$S_{n'} \subset \alpha \subset S_n$ を満たすような $\alpha \in S$ が存在しない場合、S_n は $S_{n'}$ の親であり、$S_{n'}$ は S_n の子であると定義する。この親子関係の定義に基づいて、遺伝子集合 S_n の子遺伝子集合のインデックスの集合 $C_n(1 \leq n \leq N)$ を求めた (Algorithm 1)。

Algorithm 1 MSigDB 上の遺伝子集合間の親子関係の作成方法

Input: $S = \{S_1, S_2, \cdots, S_N\}$：データベース上の遺伝子集合の集合

Output: $C_n (1 \leq n \leq N)$：遺伝子集合 S_n の子遺伝子集合のインデックスの集合

```
for n = 1, 2, ..., N do
    C_n = {}
    A = {α ∈ S | α ⊂ S_n}
    for each S_n' in A do
        if {α ∈ A, α ≠ S_n' | S_n' ⊂ α} = ∅ then
            Add n' to C_n
        end if
    end for
end for
```

遺伝子集合 S_n に含まれる遺伝子のうち、S_n の子遺伝子集合に一度も含まれない遺伝子の集合を $T_n (1 \leq n \leq N)$ とすると、

$$T_n \ = \ S_n \ / \ \bigcup_{i \, in \, C_n} S_i$$

と求められる。また、親遺伝子集合が存在しないルート遺伝子集合のインデックスの集合を r とすると、

$$r \ = \ \bigcup_{n=1}^{N} n \ / \ \bigcup_{n=1}^{N} C_n$$

と求められる。このとき、ネットワークモジュール $D_k = (V, E)$ は以下の特徴を持つ：

- $V_n (1 \leq n \leq N)$： S_n に対応したノード群。V_n に含まれるノードの数 $|V_n|$ は、S_n に含まれる遺伝子の数 $|S_n|$ を用いて、

  $$|V_n| = max(20, 0.3 \times |S_n|)$$

  と表される。
- $E_n (1 \leq n \leq N)$： V_n へ向かうエッジの集合。E_n の出発点となるノード群の集合を $I_n (1 \leq n \leq N)$ とすると、

  $$I_n \ = \ \bigcup_{i \, in \, C_n} V_i \ \cup \ \bigcup_{j \, in \, T_n} g_j$$

と表される。

あるノード群 $V_a \in I_n$ から V_n への接続は、V_a の全てのノードから V_n の全てのノードに全結合することによって行う。V_n の値は、重みを $\boldsymbol{W}$、バイアスを b とすると、インプットベクトル I_n を用いて、

$$V_n = BatchNorm(Tanh(W \cdot I_n + b))$$

と求められる。

2.2.3 output layer

2.3 評価方法

2.4 予測に用いたデータセット

第 3 章

結果

第 4 章

結論と今後の課題

第 5 章

謝辞