

各課題の進め方

1 目的

脳はどういう風に動いているのか．その気分を数理モデルを通して味わおう．ただし，講義を聞いて定期試験を受ける形式では，このレベルの力を身につけることは難しい．そこで，各課題について，各自で計算機プログラムを書いて実験をおこない，結果を図に描き，レポートを書いてもらう．単位の認定にあたっては，次のレベルを要求する（すべて口述）．

- ① 問題設定を説明できる（数式を使った定式化）．
- ② 計算機実験の手順，モデルの動作アルゴリズムを説明できる．
- ③ 計算機実験の結果を説明できる．

① ② は，メモ一枚片手に（ほぼ何もみないで）説明できることを要求する．レポート提出までたどり着いた人は，ここで示したレベルは易しい（？）はずだ．

2 基本的な注意事項

説明する相手は，モデルの振る舞いをよく知っている私なので，「理解できました！」というのを伝える程度の簡潔な説明でよい．おそらく，1課題，最速3分あれば，説明できる．ただし，「私」は，いくつもの仕事を並行して進めているので，各講義で与えた指示をすぐに思い出せるとは限らない．「先生， $\alpha = 0.1$ ではうまくいきました」と話しかけられても，「 α って何のことだっけ」と返してしまう（のはあたりまえだ）．

説明中に20秒以上黙り込まれると，こちらは困る．よくある理由は「わかっているのですが，どう説明したらいいか，わかりません」，「先生，うるさい．いまどう説明するか考えているんだから黙っててください」．こうならないよう，少なくとも一度は自分で説明の練習をしてから来ること．説明の仕方は，この講義に限らず，各講義で，新しい概念や手法がどう説明されているか，教員の説明の仕方を注意深く観察しておいてほしい．

ここで要求する口述説明は，実は，難しい．ある程度複雑で新しい知的体系を理解し他人に伝えることがいかに難しいことか，経験して欲しい．この力は，他人の話聞いて分かる力にも直結するため，知的社会で必要とされている．だからこそ，この講義では口述説明を重視する．このトレーニングから逃げていると，ろくな会社に就職できない（おどしてスマン）．これに比べると，他人の話に何かいちゃもんをつけるのは比較的易しい．友達どうし

で説明しあうなど、とことんトレーニングして欲しい。講義中には、どんどん質問してもらってかまわない。

レポートを書くためには、なにか結果を出す必要がある。結果を出すためには、計算機プログラムを書く必要がある。どういう言語で記述してもよいが、とにかく、プログラムを書くためには、与えられた課題を、把握する必要がある。レポート課題をこなすための第一歩は

1. 問題設定を理解する
2. 数理モデルの動作アルゴリズムを理解する
3. 計算機実験の手順を整理する

ことである。こういう自由度の高い？レポート課題には慣れていないせいか、何も手が進まない（何をしたらいいかわからない？）人が出現する。履修をやめたほうがいいかもしれない。しかし、くどいが、このトレーニングから逃げていると、ろくな会社に就職できない（おどしてスマン）。

何事もはじめから全体像が分かるはずもないので、完璧に理解してからとりかかろうなんてことは思わないこと。以下では、問題設定、実験手順の整理の仕方について、「例からの学習」の場合をとりあげ一例を示すが、このように整理されたものは、通常、与えられない。与えられていたとしても、自分の好みに合うように自分で整理し直すこと。自分で話を再構成することが、とても大切だ。

3 具体的な進め方

3.1 問題設定

目的：与えられた例題に対し、できるだけ正解を出力する機械を、学習により作る。

- 与えられているもの (given)

入力 $\boldsymbol{x} = (x_1, x_2)^T, x_i \in \mathcal{R}$, とそれに対する望ましい答え $y \in \{0, 1\}$ の組

$(\boldsymbol{x}^\alpha, y^\alpha), \alpha = 1, \dots, m$. 例題数 $m = 20$.

- 使用するモデル：ニューロンのモデル

$z = f\left(\sum_{i=1}^n x_i w_i - h\right)$, $n = 2$. ここで f は $0 \leq f \leq 1$ をとる単調増加関数.

ニューロンの入出力特性は、 $\boldsymbol{w} = (w_0, w_1, w_2)^T$ で決まる。各係数 w_i の初期値は乱数。

$f(0) = \frac{1}{2}$ のような関数を使う場合、 $z > \frac{1}{2}$ のとき、 $y = 1$ と判定する。

- 目的は、与えられた m 個の例題に対し、できるかぎり正解を出力するように素子の性質、つまり $w = (w_0, w_1, w_2)^T$ を変えていくこと。

与えられた課題には曖昧な点があるかもしれない。その場合は、自分で勝手に補えばよい。配布する資料に書いてあることを暗記してもしようがない。自分で納得いくまで理解しようとする。

3.2 計算機実験の手順

プログラムは、たとえば C 言語なら、main 関数を書くことから始める。関数の中身は空でよい。どういう処理をしたいか、まずはコメント文を書いていく。適宜、中身が空の関数などを追加し、書けるところから書いていく。常にコンパイルしながら進める。

1. 初期設定

係数の初期状態を乱数に設定する。例：区間 $[0,1)$ の一様乱数を生成しする ($w_i \sim [0,1)$, $i = 0, 1, 2$)。C 言語なら `srand48(7777)` と乱数の種 (7777 は例) を設定し、`w[i] = drand48();`。 `a = 0`。

2. データ (問題) の読み込み、もしくは作成

(x^α, y^α) , $\alpha = 1, \dots, m = 20$ 。具体的なデータが与えられていれば、それを読み込む。データ構造は配列 `double x[alpha][i]`, `int y[alpha]` を用意しておく ($\alpha = 0, \dots, m-1$, $i=1, 2$)。データは、自分で作ってもよい。例題の生成アルゴリズムを工夫することで、問題の難易度をコントロールできる。

3. モデルに例題 x^α を与え、出力 z を計算。

ここで、 $\alpha = a \bmod m$ 。 $z = f(u)$ 。 $f(u) = \frac{1}{1 + e^{-\lambda u}}$ のシグモイド型関数を使う。 λ はパラメータ。自分で適当な値に設定する。

4. モデルの学習： $w_i := w_i + \Delta w_i$

ここで

$$\Delta w_i \propto (y^\alpha - z)z(1 - z)x_i^\alpha, \quad i = 0, 1, 2$$

とする (\propto は比例を意味する記号)。どうしてこうするのは、別の資料で勉強すること。出力 z が、 $0, 1$ に近い値であったり、望ましい出力 y^α に近いほど係数を修正する量 (Δw_i) が少ないことがわかる。 \propto のところは、具体的には別の変数 μ などを用意し、 $\mu = 0.01$ などとあらかじめ設定し、

$$w[i] += \mu * (y[\alpha] - z) * z * (1.0 - z) * x[\alpha][i]$$

とする。

5. a++ . 2に戻り, 満足いくまで, 繰り返す.

満足いくとは, ということか. いろんな考え方があるので自分でいろいろ試して欲しい. 例えば, 「20個の例題に対し, 現状の機械がすべての例題に対し, 正解を出せるようになったら, おしまい」というのが一つの自然な方法ではある. ただし, すべての例題に対し, 正解を出すことは不可能な問題の場合, これでは無限にループが続く. 誤差

$$E = \sum_{\alpha=1}^{20} E^{\alpha}, \quad E^{\alpha} = \left[y^{\alpha} - f \left(\sum_{i=0}^2 w_i x_i^{\alpha} \right) \right]^2$$

がある程度以上小さくなればおしまいにするのが, いいかもしれない.

3.3 結果の可視化

当初の目的が理解できる, 結果が一目瞭然な図を作ってほしい. 神経回路モデルが学習する計算機実験の場合, 初期状態での機械の性能, 学習途中の性能, 学習完了後の性能など, 3通りは結果がほしい. 「例からの学習」では, 横軸に時間, 縦軸に誤差 E とする図を, 描けばよいだろう. gnuplot などで図をプロットするために必要なデータを生成する.

図の種類は1種類とは限らない. リアルタイムで, 学習が進む様子を可視化できれば楽しい. 工夫次第で理解が深まる.

3.4 動作アルゴリズムの理解

プログラムが動く楽しい. より深い理解を目指そう.

- どうして学習がうまく進むのか.
- この学習の式は, どのような理屈から導きだされたものか.
- パラメータ μ, λ には, どのような意味があるのか.
- 問題を難しく(易しく)するためには, どうしたらいいか.

どんどん自分で問題を作り, 求められている以上のことをこなそう. 卒研, 修論につながるかもしれない.