レポート課題その2 説明

- 提出は、keio.jp で
- 次の項目を忘れないで下さい
  - 学籍番号、氏名
  - 「レポート課題2」というタイトル
  - 回答そのもの
  - 感想
- 各自、独自に行ってください。レポート作成も独自に行ってください。
- レポートには実験経過や途中結果を分かりやすく（グラフ等を用いて）説明してください

- 締め切りは、2018年1月12日（金）24時です。

レポート課題 2-1

- 今回は、データ数と過学習に関する実験です。neural network を使ってみることにします（より正確には、最適なパラメータ（今回は中間素子数）の決め方の演習です）。ですから、設問は「最適な中間素子数はいくつぐらいでしょうか」となります
- まず、データを取得します（既に取得してあります）。UCI の machine learning repository 中の "Indian Liver Patient" を用いることにします。
  - インドの肝臓病患者に関するデータです。11個の属性があります。その内、Classが患者が否かを表し、値1が健常者を示します。なお、Genderは1がFemaleを表します。他の属性については、下記URLを参照してください。URL は http://archive.ics.uci.edu/ml/datasets/ILPD+%28Indian+Liver+Patient+Dataset%29
  - 各自、学習データとテストデータに分けて下さい。Upload 済みの圧縮ファイル中の「IndianLiverPatient-DataSplit.r.txt」を実行してください。Directory の設定を忘れないように。そして、学籍番号をset.seed() に用いて下さい。IndianLiverPatient.train.csv と IndianLiverPatient.test.csv が作成されます。学習データ数が、二つのクラス間で同じになるようにしています。
  - 確認のため、このプログラムが印字したものも、提出して下さい。
  - もっとも、実験するごとに、この学習データ・テストデータを作成するのならば、ファイルを作る必要はありません。

レポート課題 2-1（続）

- Neural networks を用います。
  - R の nnet を用いて学習させ、テストデータで分類精度を求めて下さい。
  - 中間素子数としては、何個が適しているでしょうか？
    - 1個から50個ぐらいの範囲で、いろいろ試してみて下さい。
  - 収束はしても、精度が悪いかもしれません。
  - これは、属性値のばらつきが大きいことが理由の一つと考えられます。そこで、各属性値（被説明変数（応答変数）を除く）の平均を0、分散を1に正規化してみましょう。Rでは、scaleという関数を用います。例えば、次のようにします。なお、学習データとテストデータを同時にscaleしなければなりません。

```
ilp.scaled <- scale(ilp[,-11])
ilp.scaled <- cbind(ilp.scaled,Class=ilp$Class)
ilp <- data.frame(ilp.scaled)
```

  - Scaleしたデータを作るプログラムIndianLiverPatient-ScaledDataSplit.r.txt を用意しておきました
  - 過学習は起こりましたか？

レポート課題 2-1（続）

- 学習データを非常に小さくしてみましょう。
  - 次のようにします。 学習データは、ilp.train に入っているとします。

```
ilp0.train <- subset(ilp.train, ilp.train$Class==0)
ilp1.train <- subset(ilp.train, ilp.train$Class==1)

n.train.small <- 20              # 学習データ数。クラス0と1は同じにした。
ilp.train.small <- rbind(ilp0.train[1:n.train.small,], ilp1.train[1:n.train.small,])
```

- 中間素子数を1個から10個ぐらいで調べてみて下さい。
  - 学習が失敗したとき（学習後、一つのクラスにしか分類しない場合。いつものようにconfusion matrix を作ってみると分かります）は、その結果は廃棄して下さい。
  - 学習結果のテスト結果（分類精度）は非常にばらつきます。何回か実験した平均値を用いて考察して下さい。

レポート課題 2-2

- 現実のデータは、思うようにいきませんね。でも、こりずに、もう一つ現実データで実験してみましょう。
- まず、データを取得します（既に取得してあります）。UCI の machine learning repository 中の "Wine quality" を用いることにします。
  - URL は http://archive.ics.uci.edu/ml/datasets/Wine+Quality です。
  - このうち の、赤ワインに関するものを用いましょう。課題2-1と同様に、各自で学習データとテストデータに分けて下さい。プログラムは、「winequality-red-DataSplit.r.txt」です。winequality-red-train.csv と winequality-red-test.csv とが作成されます。
- やはり R の nnet を用いましょう。
  - 予測すべき属性は quality です。困ったことにその値は、0～10の整数とのこと。実際には、3～8のようです。どう扱いましょうか。
  - 回帰することにしましょう。nnet の引数に linout=T を入れて下さい。

レポート課題 2-2（続）

- confusion matrix は次のようにして作ることができます。単に、nnet の予測値を四捨五入する（roundという関数を使う）だけです。nnet の結果を rw.nn に代入しているとします。

```
table(rw.train$quality, round(predict(rw.nn)))
```

- 精度の計算がちょっと面倒です。今までは正答数は「対角線」の上にある数字の和だと考えていましたが、今回は、対角線上からはずれてしまうことがあります。次のようにして下さい。confusion matrix は rw.tbl であると仮定しています。

```
correctCount <- 0
for ( i in intersect(rownames(rw.tbl),colnames(rw.tbl)) )
  correctCount <- correctCount + rw.tbl[i,i]
(accuracy <- correctCount/sum(rw.tbl))
```

- 中間素子数は1個から100個ぐらいの範囲で調べてみて下さい。一つの中間素子数に関して、数回nnetで学習させ、その平均精度を用いて下さい。
- "converged" と出たときだけ、精度を計算してください。中間素子数100個のときには、maxit=5000 とする必要があるかもしれません。
- 「以下にエラー nnet.default(x, y, w, ...) : too many (1301) weights」と言われたら、nnet の引数で、例えば、MaxNWts = 4000 として下さい。