全結合ニューラルネットワークの解釈

氏名　佐藤 怜
所属　筑波大学 情報科学類
日付　2017.1.22

1　概要

この文書では全結合ニューラルネットワーク (Fully Connected Neural Network) について得られた解釈や手法を紹介します. ここでの解釈とは, パラメータや数式からモデルの動作について何らかの人間にとって理解し易い知見を得るという意味です. モデルのパラメータや数式を解釈する目的には, 学習したパラメータからよりシンプルなデータの分離則を発見することや, モデルが重要な意思決定を担うだけの信頼に足るかどうかを調べることが挙げられます. この文書ではモデルの内部に目を向けますが, モデルをブラックボックスとしたまま解釈を行う [11] のような先行研究も存在します.

2　ノーフリーランチ定理

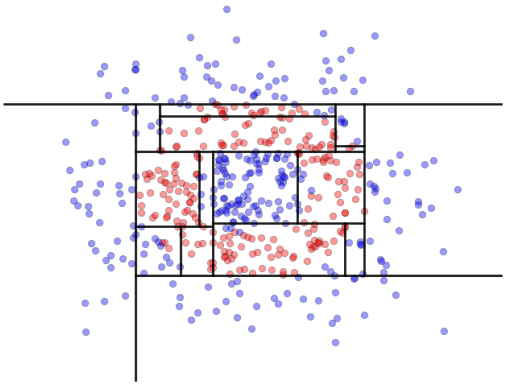

図 1　決定木の学習結果. ニューラルネットワークより決定木の方が上手く分類出来る.

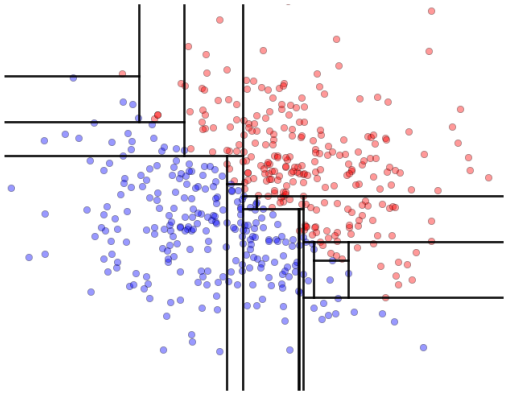

図 2　決定木の学習結果. ニューラルネットワークの方がシンプルに分類出来る.

精度の観点でのモデルの良し悪しはデータに依存します. 私はこれまでノーフリーランチ定理という言説が嫌いでしたが, 実際に幾つかの実験を行うとこのアナロジーがあながち間違いでもないことに気付きます. 図 1, 図 2に人工データに対する実験の結果を図示します.

すると結局モデルの選択は対象のデータ毎に実験を行うことで得られる解であることが分かります. (もちろん, 複数のデータセットが共通の特性を持つ場合, 1 つのモデルが複数のデータセットについて優位に立つことも有り得ます)

しかし解釈容易であるという観点から, この文書ではニューラルネットワークについて述べることにしました. 冒頭でモデルの解釈を行う目的を述べましたが, 解釈を行う対象として決定木やサポートベクトルマシンよりニューラルネットワークが適している理由を述べます.

まず決定木を考えますが, 決定木の分離超平面は入力空間の次元軸に対して常に平行です. このような 1 変数に限定した入力空間の分離は必ずしも解釈容易であるとは限りません. なぜなら同じ分離能を持つ決定木は疎でない法線ベクトルを持つ超平面分類器より分岐が多く深くなるからです. 一つ一つの分離超平面はこの上なく解釈容易ですが, 数が多ければ全体として解釈容易とは言えません.

また, 多クラス分類の美しい定式化として, カーネル関数を用いた少数のサポートベクトルマシンを組み合わせた多クラスサポートベクトルマシンが考えられます. しかし線形でないカーネル関数を用いた場合, 高次元空間での超平面分類を行いますから, これは元のデータ空間の次元での説明が困難です.

一方で, ニューラルネットワークは一見複雑ですが, 一度解釈が得られると難しいモデルではありません. 上述の 2 つのモデルの持つデメリットも持ち合わせていません. よってこの文書では多クラス分類器として実績のある全結合ニューラルネットワークを薦め, これの解釈を幾つか述べます.

記号について

この文書では, 全結合ニューラルネットワークの入力層を 0 番目の層とし, l 層 i ノードと $l+1$ 層 j ノードの間の重みを $w^l_{i(j)}$ と表記します.$b^l_{(j)}$ は $w^l_{(j)}$ に付随するバイアスです. ノード番号や次元は第 0 から数えます.

3 解釈集

3.1 分離超平面

全結合ニューラルネットワークの $l+1$ 層 j ノードの活性 h^{l+1}_j は, 活性化関数を $f(x)$ とおいて以下で与えられます.

$$h^{l+1}_j = f((\sum_i w^l_{i(j)} * h^l_i) + b^l_{(j)}) \tag{1}$$

ここで, 超平面の方程式を考えます.

$$(\sum_i w_i * x_i) + b = 0 \tag{2}$$

式 2の左辺は w と x の内積に定数を足したものです. これを $||w||||x||$ で割ると w を法線ベクトル, b を定数とする超平面からの距離になりますが, ここではノルムによる正規化はしません.

すると, 式 1の活性化関数 $f(x)$ の引数を見てみると, これは w と b をパラメータとする超平面と, ベクトル h^l_i のノルム正規化されていない符号付き距離と捉えることが出来ます.

第 3 次ニューラルネットワークブームにおいては, 活性化関数には Rectifier : $f(x) = max(0, x)$ を頻繁に用います. これにノルム正規化されていない符号付き距離を代入すると, これは超平面の上側をそのままに, 下側を 0 とおくような伝播と解釈出来ます.

よって, 任意の隠れ層のノードの活性は, そのノードへの伝播重みを法線ベクトルとする分離超平面からの距離に非線形性を与えたものと解釈します. 全結合ニューラルネットワークは 1 つの層に複数のノードが存在しますから, 同じデータ空間に複数の分離超平面を配していることになります.

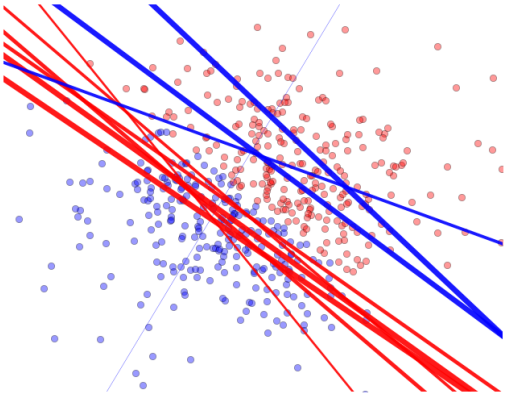

図 3　ノード数は入力層から (2,10,2).1000 個のテストデータに対して 99.3% の精度.25,000epoch.

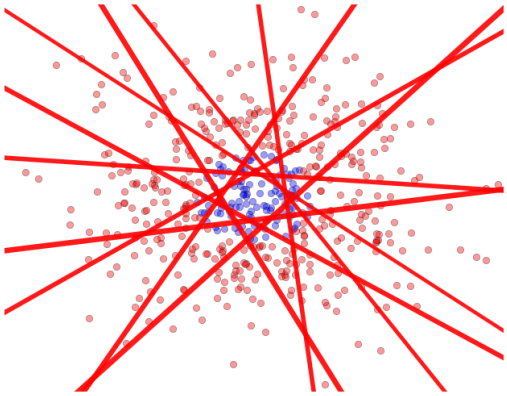

図 4　ノード数は入力層から (2,10,2).1000 個のテストデータに対して 94.3% の精度.30,000epoch.

図 3と図 4は人工的に生成したデータに対して全結合ニューラルネットワークを学習した結果の重み $w^0_{(j)}$ を分離直線として可視化した様子です. 非線形な識別を行う為には活性化関数の非線形性を利用せねばならず, 超平面は 2 クラスを仕切るように学習されます.

3.2　1-of-K 表現

K クラス識別問題において, 全結合ニューラルネットワークの教師データの表現形態として用いられる 1-of-K 表現とは, K-1 個の次元の値が 0, 残り 1 個の次元の値が 1 を取る以下のような K 次元ベクトルです.

$$t = (0, 1, 0, 0, 0, 0, ..., 0) \tag{3}$$

上記の教師ベクトルは, 1 番目のクラスが正解であるということの 1-of-K 表現です.

例えば画像ベクトルを入力として取り, そこに猫が写っているのか, 犬が写っているのか, あるいは人間が写っているのかを識別する 3 クラスの識別問題を全結合ニューラルネットワークで実装することを考えます. この時, 猫=0, 犬=1, 人間=2 と置き換えると, 猫である画像の教師ベクトルは 1-of-K 表現によって

$$t = (1, 0, 0) \tag{4}$$

と表されます.

このモデルの予測時の出力ベクトルの 0 番目の次元の値について取り上げると, これは猫である確率だったり, あるいは 1 番目の次元の値に着目してみると, これは犬である確率と捉えることが出来ます. これは人間にとって, 画像ベクトルを与えられるよりも明らかに解釈容易で有用な表現です. 全結合ニューラルネットワークの目的は, 上例の様に複雑で多様なデータ群を, 人間にとって解釈容易な表象に変換することです. またモデルの出力ベクトルに argmax を適用すればクラスを表す非負の整数が得られるので, 結局これはラベル付けと同じです.

そして前述の通り, 1-of-K 表現は確率としての解釈が可能です. 出力ベクトルが 1-of-K 表現と等しい時にエントロピーが最小となり, ある特定のクラスについて強い確信を持っている状態として解釈出来ます. エントロピーは統計力学においては乱雑さの尺度として扱われます.

以下はエントロピーの定義です.y_n はベクトル y の n 次元目の値を示します.

$$\sum_{n=0}^{K} -y_n * log(y_n) \tag{5}$$

従って 1-of-K 表現を用いた学習は, ある特定のクラスに強い確信を持つ, エントロピー最小な状態を教師とする学習です.

また, エントロピーと超平面分類器との関係は決定木の資料が参考になります.

3.3 特徴量

[13] は, 乳がんの診断法である穿刺(せんし)吸引細胞診 (針で採取したしこりの細胞の顕微鏡検査) における顕微鏡画像を元に生成した 30 次元のデータです. { 悪性, 良性 } の 2 クラス分類問題で, 0 が悪性, 1 が良性を示すラベルです. データ数は 569 で, この内 500 を学習に用い, 69 をテストに用います. 3 ノードの隠れ層を 1 層持つ全結合ニューラルネットワークを 15,000 エポックで学習します.69 のテストデータに対する予測精度は 88.4058% になりました.

ここでは, 図 5でデータに対する活性の大きさと教師ラベルの関係を可視化します.

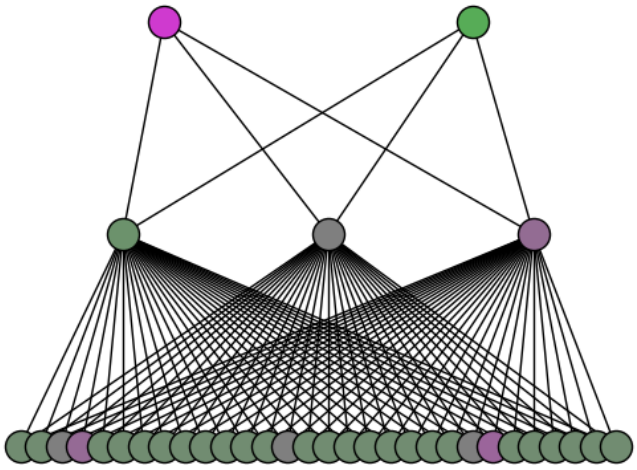

図 5　入力層第 23 ノード (最下右から 7 つ目, 紫がかった色のノード) はグラフから悪性に正の相関を持つことが分かる. この次元の値は面積の最小値を示している. こういった解釈はモデルの妥当性の検証やよりシンプルな分離則の発見に役立つ.

このグラフは学習したニューラルネットワークのアーキテクチャを示しており, 下が入力層で上が出力層で

す. このグラフを端的に説明すると, データが悪性の時にだけ活性するノードがピンク色, 良性の時にだけ活性するノードが緑色で表現されています. 鮮やかな色のノードは, 活性した時にどちらかのクラスに属している確率が高いということになります. 逆に濁った色のノードの活性は, 全てのクラスに対して弱い相関関係にあります.

この値の算出方法はプログラムとして記載しましたので, 詳細は付録を参考にして下さい.

中間層の第 0 ノード (左) は少し濁っていますが緑色に見えます. 前述の通り, このノードの活性は良性と正の相関関係にあることを意味します. 実際, 中間層から出力層への伝播重みは以下のようになっており, 中間層の第 0 ノードは出力層の第 1 ノードへ正の伝播をしています.($w^1_{(1)}$, 3.9833... のところ)

$$w^1_{(0)} = (-2.71832061, -0.46049705, 3.68458652) \tag{6}$$

$$w^1_{(1)} = (3.98333311, 0.15907529, -4.60330915) \tag{7}$$

中間層の第 0 ノードは良性へ正の伝播をしますから, 中間層の第 0 ノードに正の活性をさせるような, 具体的には $w^0_{(0)}$ との内積が大きくなるような空間が良性のデータ空間になります. このように, モデルのパラメータを介してデータが分布している多様体について述べることも出来ます. また, あるクラスに正の相関を持つノードの活性は, そのクラスの特徴量として解釈が可能です.

解釈容易な学習：ノルム正規化

学習の各エポックにおいて, 中間層の各ノードに対する入力層側の重みをまとめたベクトル, つまり $w^0_{(j)}$ のノルムを定数に正規化すると (記号の通り, 正規化するベクトルにバイアスは含みません), 中間層の活性の大きさは他のノードと概ね同じ尺度で評価出来ます. このような学習を行うと, 中間層から出力層への伝播重みは中間層のノードの重要度を示すと解釈することが可能です. 上記の例だと中間層第 0 ノードと第 2 ノードはそれぞれ出力層に大きな影響を与えますが, 濁った色の第 1 ノードは結果にあまり影響を与えていないことが分かります.

これは素朴な勾配降下法に小さく手を加えた形になりますが, この正規化を行うことでより進んだモデルの解釈が可能になります.

先に述べたように, バイアスを除く重みベクトルは分離超平面の法線ベクトルですから, これは定数倍しても分離結果に影響を与えません. (ただし活性の大きさは定数倍されます)

入力側のノルムの正規化は, 既に重み上限 (max-norm regularization) として知られています. 重み上限は入力側の重みベクトルが $||w_{(j)}|| \leq C$ を満たすように学習されます.([4] では定数 C の範囲を 3~4 としています)

[4] では, 重み上限を用いる説明として, 重みを爆発させることなく大きな学習係数を用いて最適化を高速化することが出来ると述べていますが, ここでは解釈容易性に重点を置いてノルム正規化を行いました. 尚, 重み上限は重みが $||w_{(j)}|| \leq C$ を満たす場合には重みに手を加えない点で解釈容易な学習とはやや異なります.

3.4 エントロピーピラミッド

全結合ニューラルネットワークの理想的な出力は 1-of-K です. また, 任意の深さのニューラルネットワークについて入力層から出力層に向かってエントロピーが徐々に減少し出力層で 1-of-K となるのが理想とも考えられます.

ニューラルネットワークが内部でどのような処理を行っていようとも, 出力層に向けて単調にエントロピー

が減少する, つまりあるクラスに属するだろうという確信を強めていくという過程を, ここではエントロピーピラミッドと呼ぶことにします.

仮に識別器の内部でエントロピーが増加するような層が存在すれば, それは前の層よりも複雑で, 特定のクラスに対する確信は減少していることになり, 学習に失敗している可能性があります.

ここで, 実験を行いました. 実験では, 手書き数字画像データセット [12] に対して全結合ニューラルネットワークを学習し, 精度と各層のソフトマックスエントロピーを算出しました. この実験では, 幾つかのアーキテクチャの全結合ニューラルネットワークについて, 層を重ねるごとにエントロピーが単調に減少することを確かめました.

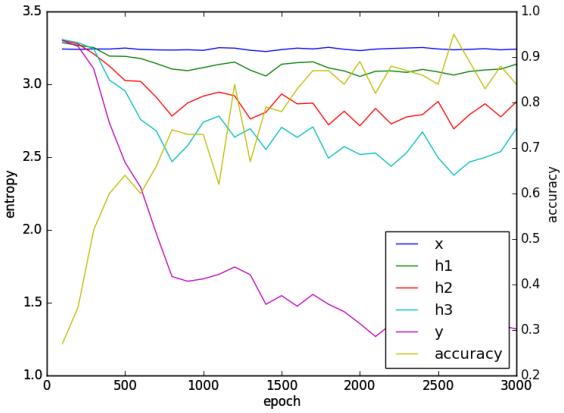

図 6　入力層から順に (784,128,64,64,10) 個のノードを持つ全結合ニューラルネットワーク

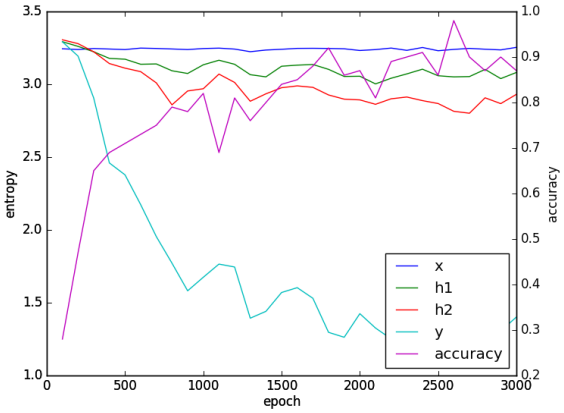

図 7　入力層から順に (784,128,128,10) 個のノードを持つ全結合ニューラルネットワーク

付録： Chainer 拡張 (Python)

ノードの活性と教師ラベルとの関連を返します.

```
class ChainerInterpret:

  def setup(self, nodes):
    self.activities = []
    num_class = nodes[-1]
    for node in nodes:
      self.activities.append(np.zeros((num_class,node)))
    self.label = []

  @classmethod
  def softmax(self, activity):
    return np.exp(activity)/np.sum(np.exp(activity))


  def watch(self, layers, t):
    label = t.data[0]
    for l in range(len(layers)):
      self.activities[l][label] += ChainerInterpret.softmax(layers[l].data[0])
    self.label.append(label)


  def fin(self):
    self.label = np.array(self.label)
    res = []

    for l in range(len(self.activities)):

      activity = self.activities[l].copy().T

      for j in range(activity.shape[0]):
        for c in range(activity.shape[1]):
          activity[j][c] /= np.sum(self.label==c)
        activity[j] /= np.sum(activity[j])
      res.append(activity)

    return res
```

使い方

```
inter = ChainerInterpret()

#ノード数を与える
inter.setup([30,3,2])

#テスト中は逐次活性を与える
x = chainer.Variable( @@ )
t = chainer.Variable( @@ )
h1 = F.relu(model.fully1(x))
y = model.fully2(h1)
inter.watch([x,h1,y],t)
```

```
#結果を取得
print inter.fin()
```

参考文献

[1] Christopher M. Bishop, パターン認識と機械学習 上

[2] Christopher M. Bishop, パターン認識と機械学習 下

[3] 岡谷貴之, 深層学習

[4] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, Ruslan Salakhutdinov, Dropout: A Simple Way to Prevent Neural Networks from Overfitting,
http://jmlr.org/papers/volume15/srivastava14a/srivastava14a.pdf

[5] John C. Platt, Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines,
https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/tr-98-14.pdf

[6] 中川裕志, 5. サポートベクターマシン,
http://www.r.dl.itc.u-tokyo.ac.jp/%7enakagawa/SML1/kernel1.pdf

[7] 福水健次, 4. サポートベクターマシン,
http://www.ism.ac.jp/%7efukumizu/ISM_lecture_2010/Kernel_4_SVM.pdf

[8] 数原良彦, SMO 徹底入門 - SVM をちゃんと実装する,
http://www.slideshare.net/sleepy_yoshi/smo-svm

[9] 櫻井彰人, 決定木 その 7 まとめ ,
http://www.sakurai.comp.ae.keio.ac.jp/classes/DENDAI/2011ML/BW/07DecisionTree.pdf

[10] Robert Tibshirani, Trevor Hastie, Margin Trees for High-dimensional Classification,
http://www.jmlr.org/papers/volume8/tibshirani07a/tibshirani07a.pdf

[11] Marco Tulio Ribeiro, Sameer Singh, Carlos Guestrin, "Why Should I Trust You?" Explaining the Predictions of Any Classifier,
https://arxiv.org/pdf/1602.04938.pdf

[12] Yann LeCun, Corinna Cortes, Chris Burges, MNIST handwritten digit database,
http://yann.lecun.com/exdb/mnist/

[13] Dr. William H. Wolberg, W. Nick Street, Olvi L. Mangasarian, Breast Cancer Wisconsin (Diagnostic) Data Set,
https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)