

# DNNレポート

伊原 康行

---

## Day1 NN (ニューラルネットワーク)

- 全体像

- ニューラルネットワークは、入力層・中間層・出力層から成る数理モデルであり、回帰分析（連続値を出力する関数の近似）、分類（性別などの離散的な結果を予想）が出来る。

- 1. 入力層～中間層

- 入力層の各成分に重みパラメータを乗算したものを総和し、これを中間層の入力とする。ニューラルネットワークの学習では、重みパラメータの最適化を目的とする。

- 2. 活性化関数

- ニューラルネットワークの各層(入力層は除く)において、入力から次の層への出力を決める非線形関数を「活性化関数」と呼ぶ。具体的には①ステップ関数、②シグモイド関数、③RELU関数がある。
  - ①は出力値が0 or 1のみのため、微分不可能な点がある、線形分離可能なものしか学習できないなどの課題があった。
  - ②は出力値が0～1の間の連続値を取り得るため、全ての入力において微分可能、①より解ける問題の幅が広いなどの便利な点があるが、出力値が1未満が多いため、多層ニューラルネットワークで重みを最適化する問題を誤差逆伝播法で解こうとした際に、勾配消失するという課題があった。
  - ③は、多層ニューラルネットワーク（ディープラーニング）で使われるもので、勾配消失問題の回避とスパース化による学習の効率化などのメリットがある。

- 3. 出力層

- 誤差関数...ニューラルネットワークの性能を評価するために、正解と出力との差の期待値を表す「誤差関数」が定義される。誤差関数は、回帰問題、分類問題のそれぞれにおいて、二乗誤差、交差エントロピーの形で定義され、誤差関数を小さくする様にニューラルネットワークの学習（重みパラメータの最適化）が行われる。
- 活性化関数...出力層の活性化関数は、回帰分析の場合は恒等写像、分類問題（2クラス）の場合はシグモイド関数、分類問題（多クラス）の場合はソフトマックス関数が用いられる。

- 4. 勾配降下法

- 勾配降下法...ニューラルネットワークでは、誤差関数を最小化する重みパラメータを見つけるために、誤差関数の微分係数を用いて重みパラメータを逐次的に更新する「勾配降下法」が利用される。収束性の向上のために学習率の調整をする必要がある。ただし、微分係数は学習サンプル全てを用いて計算されるため計算コストが大きいというデメリットがある。
- 確率的勾配降下法(SSD)...勾配降下法における微分係数の計算コストを緩和するために、ランダムに抽出したサンプル(1個)のみを用いて重みパラメータを逐次更新するのが確率的勾配法である。

- 。ミニバッチ勾配降下法...学習サンプル全体をランダムに分割したサンプル集合を用いて、重みパラメータを更新するのがミニバッチ勾配降下法である。

#### 5. 誤差逆伝播法

- 。ニューラルネットワークの層数が多くなると、勾配降下法における微分係数が増えてしまい、重みパラメータを更新するのが大変になる。この課題を解決した効率的な計算法が誤差伝播法である。具体的には、最終層で算出された誤差を用いて、出力層に近い層から順次、微分係数を求める。

## Day 2 CNN (畳み込みニューラルネットワーク)

### ・ディープラーニングの学習テクニック

#### 。勾配消失問題

- 多層のニューラルネットワークの時に、誤差逆伝播の影響で下位層（入力層に近い層）に近づくにつれ、勾配(誤差関数の重みパラメータでの微分)がどんどん緩やかになっていく。そのため、重みパラメータが更新されにくくなるという問題。この問題を回避するために、次の3つが考えられている。

1. 活性化関数の選択...従来使われることが多かったシグモイド関数は、出力値が0~1の連続値のため、誤差逆伝播を重ねることで勾配が緩やかになりがちだったが、代わりにReLU関数を使うことで問題を回避
2. 重みの初期値設定...重みの要素を、前の層のノード数の平方根で除算した値などを初期値に用いる。
3. バッチ正規化...入力データの偏りをなくすために、中間層の活性(入力値)の成分ごとに、正規化(ミニバッチ単位で平均0、分散1)を行う。

#### 。学習率最適化

- 勾配降下法で重みパラメータを最適化する際、学習率の値が大きいと最適値にいつまでもたどりつかず発散する、逆に値が小さいと収束するまでに時間がかかる、という問題がある。この問題を回避するため、モーメントを用いた「モーメント法」、過去の勾配の情報を利用して学習率を最適化する(最適解に近づくにつれ学習率を小さくする)「AdaGrad」などがある。AdaGradでは鞍点から抜け出せないという問題があるため、過去の勾配の情報を減衰させた「RMSProp」、RMSPropにモーメント法の考え方を加えた「Adam」が考案されている。

#### 。過学習

- 特定の訓練サンプルに特化して学習してしまう問題を「過学習」という。この場合、訓練誤差は小さくなるが、未知のテストデータに対するテスト誤差は大きくなる。この問題を回避するために次の手法が考えられている。
  - L1正則化、L2正則化...誤差関数にpノルムを加えて重みパラメータの大きさを抑制しながら学習する方法をLp正則化(p=1,2)と言う。L1正則化はスパース解(0値の成分が多い重みパラメータが得られる)を得やすいという特徴がある。
  - ドロップアウト...多層ニューラルネットワークではノードの数が多い(従って重みパラメータの数も多い)ため、過学習が起こりやすい。そこで学習を繰り返すたびに、ランダムにノードを削除させて学習させ、過学習を回避する手法を「ドロップアウト」と言う。

### ・畳み込みニューラルネットワーク

- 畳み込み層

- カラー画像を入力値とする場合、画像内の物体の形状や色彩の特徴を上手にとらえるために、フィルターを重みに用いて中間層の入力を得る「畳み込み」の手法が考えられている。畳み込みの演算概念として、多層になるほど出力のサイズが小さくなるのを回避する「パディング」、フィルタをずらす間隔を指定する「ストライド」、入力データのパターン数(RGB 3色など)を表す「チャンネル」がある。

- プーリング層

- 入力画像の(若干の)位置ずれに対するロバスト性を確保するために、対象領域の最大値または平均値を取得するプーリング層が導入されている。

## Day3 RNN (再帰的ニューラルネットワーク)

- 音声データ、テキストデータなどの時系列データを扱うために、RNN(再帰的ニューラルネットワーク)がある。初期の状態と過去の時間( $t-1$ )の状態を保持し、そこから次の時間での $t$ を再帰的に求める再帰構造がRNNの特徴である。通常のニューラルネットワークにループ構造が入ったものと考えれば良い。
- BPTT...時刻 $t$ のRNNの隠れ層には、前の時刻( $t-1, t-2, \dots$ )の影響もあるため、誤差逆伝播法で重みパラメータを更新する際は、時刻をさかのぼって更新する必要がある。RNNを時刻で展開すると通常のNN(順伝搬型NN)とみなせるので、そこで通常のNNと同様の誤差逆伝播を行い、重みパラメータを更新する。
- LSTM...BPTT手法の導入により誤差伝播法を用いて重みパラメータを更新することはできるが、多層ニューラルネットワークと同様、勾配消滅問題や、勾配爆発問題が起こりやすい。その解決策としてCECが考案されているが、CECは、過去の情報が全て保管されており、時間依存度が考慮されていない。そこで過去の情報の保管期間をコントロールするため、入力ゲート、出力ゲート、忘却ゲートを導入して、情報の保管期間をデータから学習してしまう手法が「LSTM」である。
- GRU...LSTMはパラメータが多く、計算負荷が大きいという課題がある。そうした問題を解決するために、もっとシンプルな方法でLSTMと同じ性質、精度を確保するようにしたものが「GRU」である。
- Seq2Seq...RNNを用いたEncoderDecoderモデル(自然言語処理において文章を入力として文章を出力するモデル)の一種であり、機械対話や機械翻訳などに使用されている。

## Day4 強化学習

- 「強化学習」
  - エージェントが報酬を最大化できるように、環境の中で行動を選択する方法を学習。
  - 優れた方策を見つけることが目標であり、教師付き学習、教師無し学習とは目標が異なる。
- 「行動価値関数」...状態と行動のペアに対する価値に着目した関数
- 「方策関数」...方策ベースの強化学習において、ある状態である行動をとる確率を与える関数( $p(\text{行動} \mid \text{状態})$ の事後確率の形)
- 「方策勾配法」...方策関数のパラメータを、勾配法を用いて最適化する(報酬が最大化する様なパラメータを探索する)方法。「方策勾配定理」を用いて、「報酬の勾配」を「方策の勾配」と「行動価値関数」のみを用いて簡潔に表現できるため、計算がシンプルになる。