深層学習(MLP) 3 章_前半

確率的勾配降下方

3.1勾配降下方

順伝搬型ネットワークおさらい

順伝搬型ネットワークの学習では訓練データDの入力ベクトルに対してパラメータwの重み付けを行うことでユニットに対しての入力値が得られ、それに活性化関数を適用することで出力が得られる。そして訓練データDの結果データと計算結果により得られた出力の誤差を誤差関数により求められる。

訓練データ： $D = (x_1, d_1), ..., (x_n, d_n)$

誤差関数(2値)： $E(w) = \frac{1}{2}\sum_{n=1}^{N} ||d_n - y(x_n; w)||^2$

誤差関数(多クラス)： $E(w) = -\sum_{n=1}^{N}\sum_{k=1}^{K} d_{nk} \log y_k(x_n; w)$

3.1勾配降下方

機械学習では順伝搬型ネットワークによる誤差E(w)が小さくなることがゴールである。

E(w)が一番小くなる値を見つけたくても、一般に極小値は多数存在し最小値を見つけるのは難しい

→E(w)がある程度小さい値を見つけるのであればwを繰り返し更新することにより求められる

3.1勾配降下方

勾配降下方

wの更新方法で最も単純なのは勾配降下方である。wを更新する方向(勾配)は以下になる

$$\nabla E = \frac{\partial E}{\partial w} = \left[\frac{\partial E}{\partial w_1}, , , \frac{\partial E}{\partial w_M} \right]^T$$

勾配降下方ではwを負の勾配方向(-∇E)に少し動かすのを繰り返し実行する。wの値は以下のように更新される。

$$w^{(t+1)} = w^{(t)} - \epsilon \nabla E$$

Wの更新方法には他には目的関数の2次微分を利用するニュートン法などより高度なものがあるが、ディープネットの場合は目的関数の2次微分が複雑になり勾配降下ほうの方が有効になる

3.2確率的勾配降下法

□バッチ学習

バッチ学習では訓練データを全て使って誤差Enの和が誤差関数となっていた

$$E(w) = \sum_{n=-1}^{N} E_n(w)$$

□確率的勾配降下法

確率的勾配降下法(SGD)では訓練データの一部を使って誤差を評価する。一回学習が済んだらまた別のデータを使い、毎回学習用データを選び直すようにする。

確率的勾配降下方にはバッチ学習に比べ以下のメリットがある。

- 学習データに冗長性がある場合は、学習結果に影響せず素早く学習できる
- 望まない極小値にはまってパラメータが更新できなくなるリスクを低減できる
- パラメータの更新が小刻みに行われるため、学習過程がわかりやすい
- データ収集と学習を同時並行に行えるためオンライン学習が行える

3.3ミニバッチ

パラメータ更新を1つのサンプル単位で行うのではなく、複数のサンプルで同時に行う時このサンプル集合をミニバッチと呼ぶ。

Nt個のサンプルがあるミニバッチDtの全サンプルに対する誤差を以下とし

$$E_t(w) = \frac{1}{N_t} \sum_{n \in D_t} E_n(w)$$

勾配の方向にパラメータを計算する。

ミニバッチのサンプル数は大体10~100

　普通は分類するクラス数と同じサンプル数が用いられる

　クラス数がそれより多い場合、クラス数と同一のサンプル選択後ランダムに選択していミニバッチに適用する

　複数のサンプルで並列に学習を進めることができる