

※自身が知らなかった知識を中心にレポートをまとめている

深層学習 day2

Section1:勾配消失問題

○勾配消失問題

- ・ 中間層を増やしていくと、うまく誤差が伝播されなくなることがある
- ・ 0~1の値を微分していくと勾配の値が小さくなっていく
- ・ シグモイド関数：微分すると最大値が0.25になってしまう(入力値が0の時)

○解決方法

- ・ 活性化関数の選択
- ・ 重みの初期値設定
- ・ バッチ正規化

○活性化関数の選択

- ・ ReLU 関数：勾配消失問題の回避とスパース化に貢献
値が0より大きい場合、微分すると1になるような関数
微分値が0になる時、必要な場所だけ抽出できるのでスパース化できる

○重みの初期値設定

- ・ Xavier の初期値: Sigmoid 関数など活性化関数に線形関数を用いるとき
標準正規分布の重みを前の層のノード数の平方根で除算した値 ($1/\sqrt{n}$)
- ・ He の初期値: ReLU 関数など活性化関数に非線形関数を用いるとき
標準正規分布の重みを前の層のノード数の平方根で除算し $\sqrt{2}$ をかけた値 ($\sqrt{2}/\sqrt{n}$)

○バッチ正規化

- ・ バッチ正規化：ミニバッチ単位で入力値のデータの偏りを抑制する手法
一学習の安定化による学習の高速化、過学習の抑制、初期値にそれほど依存しなくなる、などの効果
- ・ 活性化関数に値を渡す前後に、バッチ正規化を挟む

Section2:学習率最適化手法

○Momentum

- ・ 誤差をパラメータで微分したものと学習率の積を減算した後（勾配）、現在の重みに前回の重みを減算した値と慣性の積を加算する
- ・ 前回の重みの大きさが運動量的なものに相当する
- ・ 局所的最適解にならず、大域的最適解となる

○AdaGrad

- ・ 誤差をパラメータで微分したもの（勾配）と、再定義した学習率の積を減算する
- ・ 勾配の緩やかな斜面に対して最適解に近づける

- ・学習率が徐々に小さくなるので、大域的最適解に辿り着かず鞍点問題を引き起こす

○RMSProp

- ・誤差をパラメータで微分したもの（勾配）と、再定義した学習率の積を減算する
- ・AdaGradの学習率の再定義方法を修正し、前回までの勾配情報を活かすかのパラメータを追加
- ・局所最適解にならず、大域的最適解になる

○Adam

- ・MomentumとRMSPropのメリットを組み合わせたもの

Section3:過学習

○全体像

- ・原因
 - （・訓練データが少ない）
 - ・パラメータの数が多い（モデルの表現力が高い）
 - ・ノードの数が多い
 - ・パラメータの値が適切ではない、など
- ネットワークの自由度が高い

○正則化

- ・ネットワークの自由度（層数、ノード数、パラメータの値など）を制約すること
→正則化手法を利用して過学習を抑制する
- ・L1正則化（ラッソ回帰）、L2正則化（リッジ回帰）、ドロップアウト
- ・過学習の原因：重みが大きい値をとってしまう
→誤差に対して、正則化項を加算することで重みを抑制する。
Weight decay(荷重減衰)
- ・pノルムが1（マンハッタン距離）：L1正則化（スパース性がある）
pノルムが2（ユークリッド距離）：L2正則化

○ドロップアウト

- ・ランダムにノードを削除して学習させること

Section4:畳み込みニューラルネットワークの概念

○CNN

- ・CNNでは次元間でつながりのあるデータを扱える
カラー画像(x, y, RGB)、動画(時刻(Frame), x, y, RGB)

○畳み込み層

- ・入力値 * フィルター → 出力値 + バイアス → 活性化関数 → 出力値
フィルターの数に応じて出力値も同じ数できる
次元の繋がりを保った特徴を学習

画像の場合、縦、横、チャンネルの3次元の繋がりを保った学習

- ・全結合層での学習
これまでのニューラルネットワークのような学習を最後に挟む
- ・出力値に対応してバイアスを用意する

○パディング

- ・入力画像にフィルターをかけると、出力画像が小さくなってしまう
畳み込み演算を繰り返すと、どんどん出力画像も小さくなる
→パディングを行い、出力画像のサイズを揃える
- ・パディングの方法
0埋め、近い数字など

○ストライド

- ・フィルターのストライド1ではなく2などに設定することができる
- ・ストライドを大きくすると、出力サイズが小さくなる

○チャンネル

- ・フィルターの数を増やし、チャンネル方向も含めた3次元データで畳み込み処理を行える

○プーリング層

- ・プーリング層は重みがない
- ・対象領域のMax 値 or 平均値を取得

Section5:最新のCNN

○AlexNet

- ・5層の畳み込み層およびプーリング層など、
それに続く3層の全結合層で構成
- ・ImageNetデータセットを学習させた
- ・全結合層の出力にドロップアウトを使用（過学習防止）
- ・畳み込み層 → 全結合層の処理
3次元の畳み込み層を、1次元に置き換える（Flatten）
その他に Global Max Pooling, Global Average Pooling などの手法
：各チャンネルごとに最大/平均の値を使う（256チャンネルだと256個になる）