

機械学習・数理分野コーディング課題レポート

東京大学工学系研究科システム創成学専攻 武田惇史

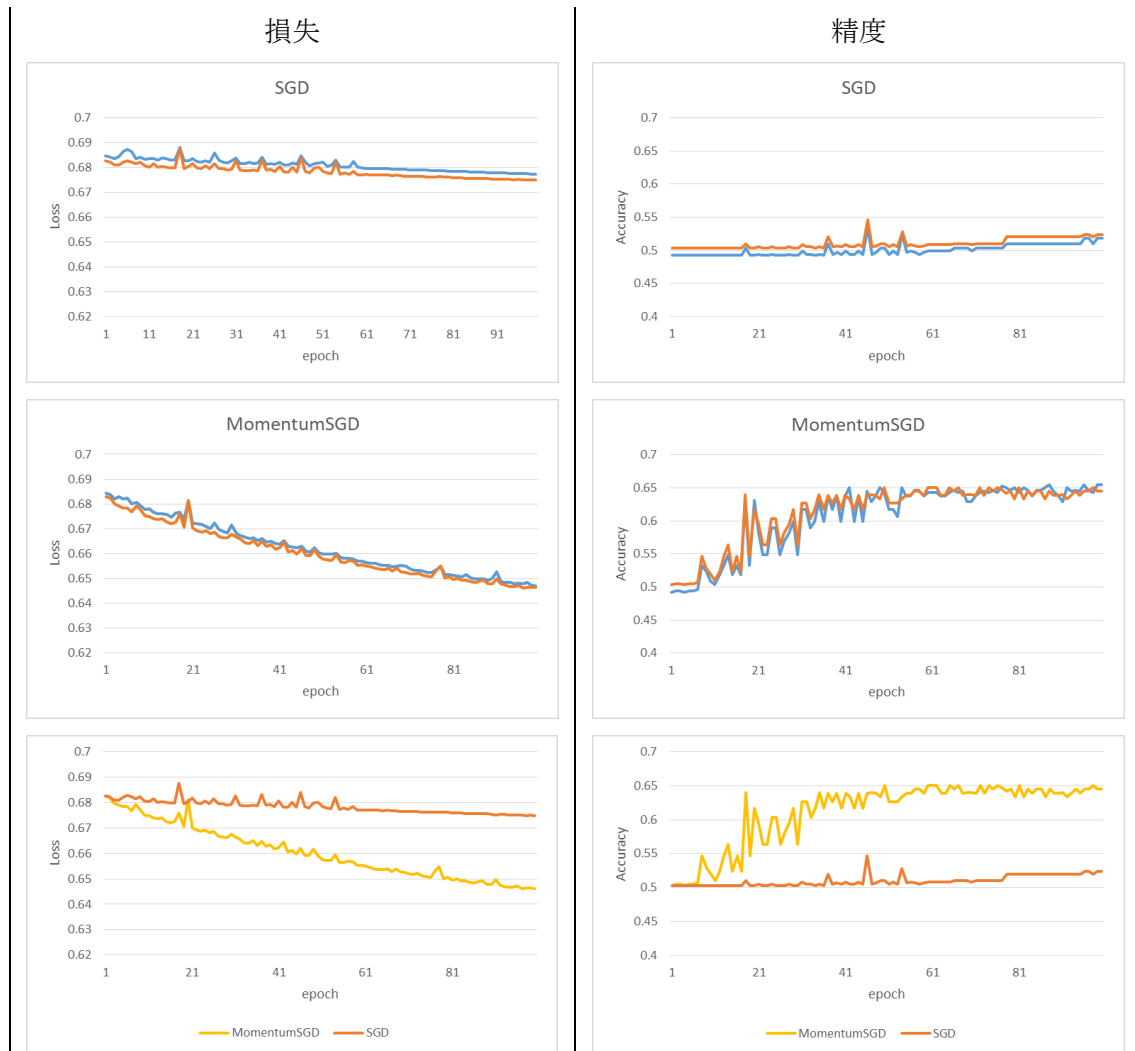
課題 3

与えられたデータに対し、SGD 及び MomentumSGD による学習を行い、損失と精度について比較する。各パラメータは、以下のように設定した。

- ・特長量ベクトルの次元 $B = 8$ 、ステップ数 $T = 2$
- ・学習率 $\alpha = 0.0001$ 、モーメント $\eta = 0.9$ 、数値微分の摂動 $\varepsilon = 0.0001$
- ・エポック数：100

以下の図では、青色の線が訓練データ、オレンジ色の線が検証データに対する値を表す。

下二つの図は検証用データに対する各値の比較である。



MomentumSGD によって収束が速くなっていることが分かる。MomentumSGD を使った場合、精度はおよそ 65%に達した。

課題 4

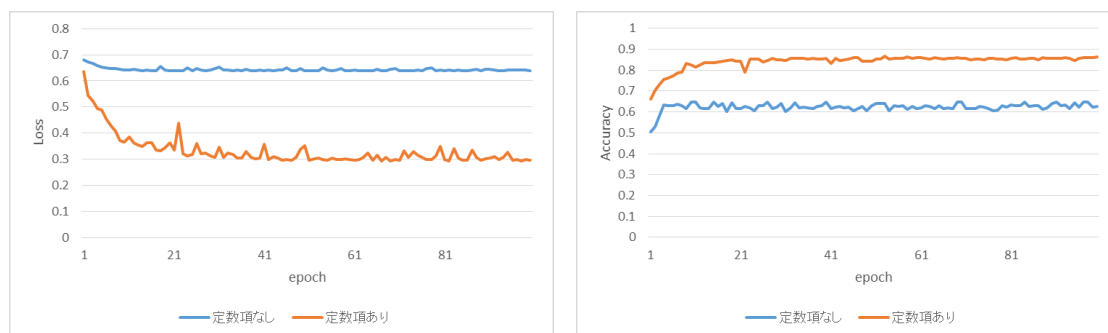
課題 3 のモデルには問題がある。GNN の出力である特徴量ベクトルは、入力グラフの違いによって定数倍の違いしかない。したがって、 D を大きくしても実質的な次元数は 1 であり、モデルとしての表現力は非常に低い。これは、ReLU が乗法的関数であるが故に起こるものである。

この問題を回避するため、(集約-2)の直前に、ベクトル a の末尾に定数項(1)を追加した。これに伴い、行列 W のサイズは $D \times (D + 1)$ となる。

パラメータを以下のように変え、定数項の有無で比較する。

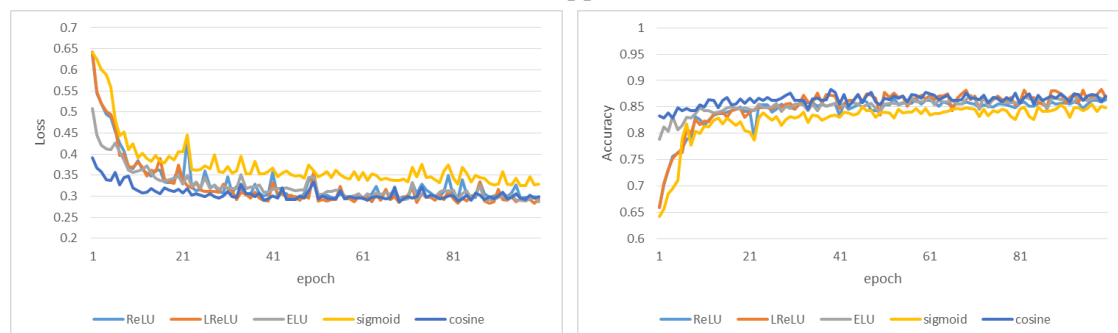
- $D = 8, T = 3$

- 最適化器は Adam とする。(この変更による改善についてはこのレポートでは省略する。このあと複数の活性化関数で比較を行う際、パラメータに関してロバストな最適化器が必要だった。) Adam のパラメータは参考文献[1]中のデフォルト値を採用。



定数項を付け加えたことで、モデルの表現力が上がり、精度が 85% 程まで改善した。

さらに、定数項を付け加えたモデルにおいて、活性化関数を変えたときの損失、精度を調べた。各活性化関数の実装は `src/Functions.hpp` にある。



最も収束が遅かったのは sigmoid 関数であった。モデルのステップ数 $T=3$ と深くしたことから、sigmoid 関数特有の勾配消失が起こっているものと考えられる。

sigmoid 関数以外には大きな差は現れなかった。今回実験した限りでは最も収束が速いのは cosine 関数であったので、test データに対する予測では、活性化関数に cosine 関数を使用したものを用いた。

参考文献

[1] Kingma, Diederik P., and Jimmy Ba. "Adam: A method for stochastic optimization." ICLR 2014.