

## 進捗報告

### 1 今週やったこと

- 2 クラスと 4 クラスの分類

### 2 問題設定

データを正解と不正解に分割して学習する以下のアルゴリズム 1 を考える.

---

#### Algorithm 1 Swap two datasets

---

1. データ 2N 抜き出す
  2. そのデータを ランダムに N (A) と N (B) に分ける.
  3.  $X : A \rightarrow \text{train}$ ,  $B \rightarrow \text{test}$  で学習
  4.  $Y : B \rightarrow \text{train}$ ,  $A \rightarrow \text{test}$  で学習
  5. 3. と 4. で test でミスしたデータを調べる
  6. A をテストとしたとき失敗したデータと B をテストとしたとき成功したデータを入れ替える. ただし, 同じクラスのデータ同士.
  7. モデルを初期化して, 3. に戻る.
- 

データのクラスが同じもの同士を入れ替えることで, データセットのクラス比が変化しないようにした. 学習中に入れ替えるのではなく, 学習が終わった後入れ替え, 再び学習する際にモデルのパラメータは初期化した.

### 3 実験

今回は cifar10 に含まれる 10 クラスの内, 2 クラスの場合と 4 クラスの場合の簡単な分類タスクを用意して, 前回の 10 クラスに近い条件で同様の実験を行った.

#### 3.1 結果

前回よりもクラス数が減ったため, ベースラインも上昇した. それと共に図 1 のように高い精度が実現でき

表 1: 実験の設定

dataset	cifar10
data N	5,000 / model
task	2 or 4 クラス識別
input	image(3x32x32)
output	class(2 or 4)
model	CNN(16 層)
optim	SDG (lr=0.001, moment=0.9)
loss	Cross Entropy Loss
batch size	16
epoch	20
swap	15

た. 単に CNN の出力層を変えただけであるため, 実行時間はほとんど変化なく 90 分程であった. epoch 数を増やすことはできなかったが 20 epoch 時点の精度で十分であると判断した.

前回 10 クラスで精度は 2 モデルとも似た推移をとったが, 2 クラス, 4 クラス共に手順 4 のモデルの精度が高い性能を示した. これはアルゴリズムから予想通りで, 前回と比べて十分に精度が出る問題に設定したため, 2 つのデータセット間の入れ替えでテスト結果が強く反映されたと考えられる.

図 2 では手順 6 でのデータ入れ替え数を示した. 入れ替えの多くが鳥 (bird) クラスに占められており, 逆に自動車 (automobile) クラスは入れ替えが少なかった.

さらに入れ替えを調べるため, 図 3 には入れ替え頻度分布を示した. 実験では 15 回の交換を行ったため, あるデータが頻繁に入れ替えられたときグラフでは 15 回に近い値をとる. 自動車は交換されたものはほとんどが 1 回なのに対し, 鳥は何度も入れ替えられていることが分かった. 自動車が簡単な分類であるため, モデルのテスト結果が変化せず, 入れ替えが起こりにくかったと思われる.

### 4 考察

前回とほとんど変化のないコードで大きく異なる結果となったのは, 問題の難しさに対して前回のモデルが

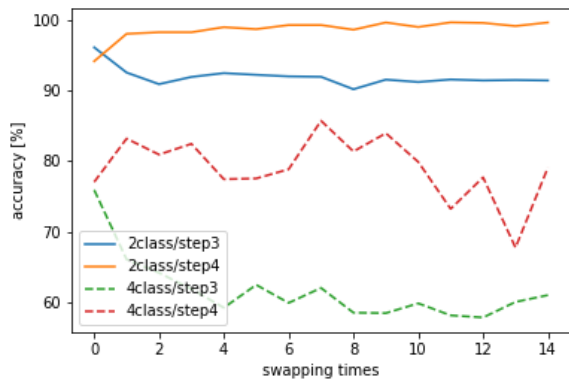


図 1: 学習後モデルのテストの精度 実線が 2 クラス分類, 点線が 4 クラス分類

十分に機能していなかったことが考えられる。問題を簡単にすることによって、データセットの特性が顕著になったと言える。

頻度分析をすることで交換されるデータの流れやクラスごとの違いが分かった。またテスト精度をクラスごとに分けることで、鳥クラスが何度も入れ替えられた原因を探りたい。

## 5 今後の予定

- 精度を高めるためのモデルの調整

## 6 ソースコード

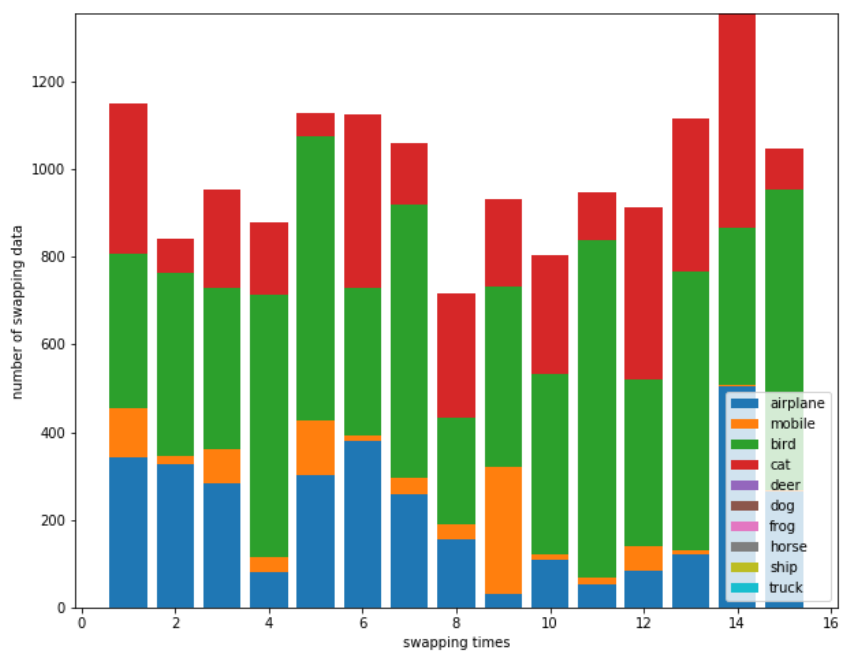


図 2: 4 クラス分類でのデータ入れ替え数

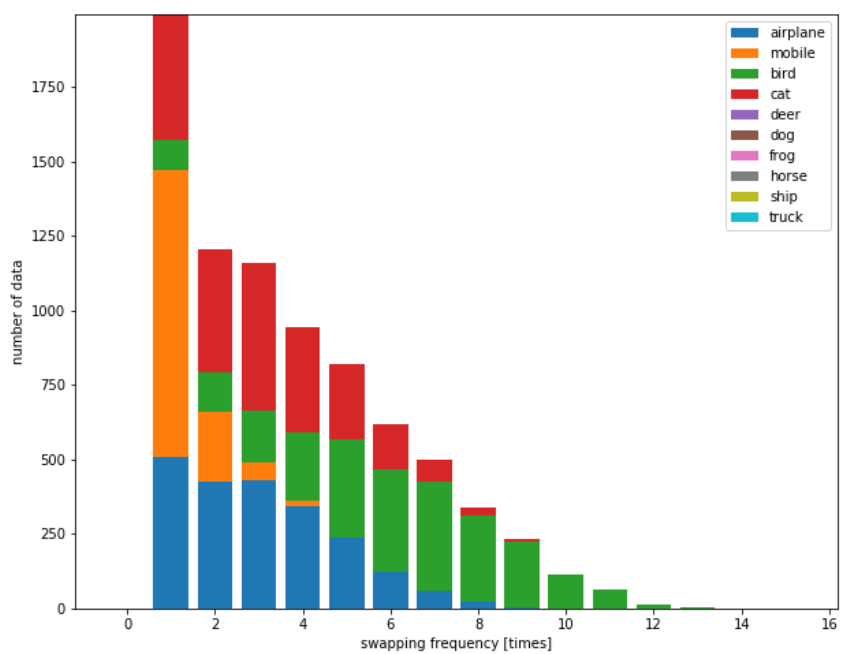


図 3: 4 クラス分類でのデータの入れ替え頻度分布 ( $0 \leq$  同じデータが入れ替えられた度数  $\leq 15$ )