

2. 画像認識

cifar-10.zipを解凍して、データセットとして用いよ、

データセットを使いこなそう

1. class.csvにクラス情報がある。label（数字）とクラス名を表示せよ。
2. train.csvは学習データの情報が記載してある。学習データ数はいくつか？順番に画像を表示せよ。その際、タイトルにファイル名と、クラス名とラベル（数字）を表示せよ。
3. ランダムに学習画像を表示せよ。
4. チュートリアル；[Create a dataset class](#)に倣って、Datasetクラスを作成せよ。また、データとラベルを取得し、画像とラベルを表示するクラス関数（show_datapoint()）も作成せよ。
5. TorchvisionにはPIL画像を変換するモジュールがある。Datasetから取得した画像に対して、128画素四方にリサイズ、ランダム回転（0-90度）する変換を施してみよ。[Data Transoformation](#), [Compose](#), [Resize](#), [RandomRotation](#), [RandomHorizontalFlip](#), [Array->Image](#)の変換, Image -> Arrayは asarray or array.
6. データセットクラスに変更を加えて、画像取得時に変換を加えよ。ヒント：具体的には、クラス変数に画像変換のクラスを登録し、データ取得時に変換を加えます。
7. [Dataloderクラス](#)を使って、ミニバッチを出力するデータローダーを作成せよ。得られたミニバッチの画像をまとめて表示せよ（ついでにラベルも）。
8. ニューラルネットワークの訓練に使用するために、ミニバッチの画像データを、（1）Tensorに変換する、（2）正規化する必要がある。これを実現するよう変更せよ。[ToTensor](#), [Normalize](#). 学習データから、平均と標準偏差を計算してね（画像変換なしの画像から計算してね）。ミニバッチの平均0、標準偏差1に近くなれば、OKです。

モデルを作ろう

課題

9. [VGG \(Configuration A, B, D, E, w/wo BathchNorm\)](#)を作成せよ。
10. [ResNet-18, 34, 50, 101, 152](#))を作成せよ。

ヒント1 (VGG)

- 全てのconvは、カーネルサイズ 3x3, パディング 1, ストライド 1.
- conv層の次に、BatchNorm (オプションで、ありの場合, なしの場合がある), ReLUがきます.
- Max poolingは、カーネルサイズ 2x2, ストライド 2.
- Dropoutは 2 つの全結合層の直後に適用します. $p=0.5$.
-

ヒント2 (ResNet)

- 基本的な設定は, Table 1, Fig. 3, Fig. 5に記載されている.
- ほとんどのconv層は3*3, padding=1, stride=1. conv層の直後にBatchNormがある.
- 特徴マップをダウンサンプリングする際は, conv層のstride=2を用いる.
- ショートカットのダウンサンプリングは, 1x1のconv層でstride=2を使う.
- まずは, Basicブロックから始めるといいかも

学習してみよう

11. train.csvを使って, 学習してみよう.

テストしてみよう

12. test.csvを使って, テストしてみてね.