

1. Download CIFAR-10 Python version link from :

<https://www.cs.toronto.edu/~kriz/cifar.html>
[\(https://www.cs.toronto.edu/~kriz/cifar.html\)](https://www.cs.toronto.edu/~kriz/cifar.html)

We will be using CIFAR -10 data set (from Canadian Institute For Advanced Research) for classifying images across 10 categories

```
In [1]: # Put file path as a string here.  
CIFAR_DIR = 'C:\\Users\\Sreekanth\\Documents\\Datascience\\cnn\\cifar-10-batches-py\\'
```

2. Import the CIFAR dataset

Pickle is used for serializing and de-serializing a Python object structure. Any object in python can be pickled so that it can be saved on disk. Pickle does is that it “serialises” the object first before writing it to file. Pickling is a way to convert a python object (list, dict, etc.) into a character stream. The idea is that this character stream contains all the information necessary to reconstruct the object in another python script.

```
In [2]: def unpickle(file) :  
    import pickle  
  
    with open(file, 'rb') as fo:  
        cifar_dict = pickle.load(fo, encoding='bytes')  
  
    return cifar_dict
```

```
In [3]: dirs = ['batches.meta', 'data_batch_1', 'data_batch_2', 'data_batch_3', 'data_batch_4' , 'data_batch_5', 'tes  
t_batch' ]
```

```
In [4]: all_data = [0, 1, 2, 3, 4, 5, 6]
```

```
In [5]: for i, direc in zip(all_data, dirs) :  
    all_data[i] = unpickle(CIFAR_DIR + direc)
```

```
In [6]: batch_meta = all_data[0]  
data_batch1 = all_data[1]  
data_batch2 = all_data[2]  
data_batch3 = all_data[3]  
data_batch4 = all_data[4]  
data_batch5 = all_data[5]  
test_batch = all_data[6]
```

```
In [7]: print(data_batch1.keys())  
  
dict_keys([b'batch_label', b'labels', b'data', b'filenames'])
```

3. Reading the label names

```
In [8]: batch_meta
```

```
Out[8]: {b'num_cases_per_batch': 10000,  
         b'label_names': [b'airplane',  
                           b'automobile',  
                           b'bird',  
                           b'cat',  
                           b'deer',  
                           b'dog',  
                           b'frog',  
                           b'horse',  
                           b'ship',  
                           b'truck'],  
         b'num_vis': 3072}
```

In [9]: data_batch1

```
Out[9]: {b'batch_label': b'training batch 1 of 5',
  b'labels': [6,
  9,
  9,
  4,
  1,
  1,
  2,
  7,
  8,
  3,
  4,
  7,
  7,
  2,
  9,
  9,
  9,
  3,
  2,
  6,
  4,
  3,
  6,
  6,
  2,
  6,
  3,
  5,
  4,
  0,
  0,
  9,
  1,
  3,
  4,
  0,
  3,
  7,
  3,
  3,
  5,
  2,
```

2,
7,
1,
1,
1,
2,
2,
0,
9,
5,
7,
9,
2,
2,
5,
2,
4,
3,
1,
1,
8,
2,
1,
1,
4,
9,
7,
8,
5,
9,
6,
7,
3,
1,
9,
0,
3,
1,
3,
5,
4,
5,
7,

7,
4,
7,
9,
4,
2,
3,
8,
0,
1,
6,
1,
1,
4,
1,
8,
3,
9,
6,
6,
1,
8,
5,
2,
9,
9,
8,
1,
7,
7,
0,
0,
6,
9,
1,
2,
2,
9,
2,
6,
6,
1,
9,

5,
0,
4,
7,
6,
7,
1,
8,
1,
1,
2,
8,
1,
3,
3,
6,
2,
4,
9,
9,
5,
4,
3,
6,
7,
4,
6,
8,
5,
5,
4,
3,
1,
8,
4,
7,
6,
0,
9,
5,
1,
3,
8,

2,
7,
5,
3,
4,
1,
5,
7,
0,
4,
7,
5,
5,
1,
0,
9,
6,
9,
0,
8,
7,
8,
8,
2,
5,
2,
3,
5,
0,
6,
1,
9,
3,
6,
9,
1,
3,
9,
6,
6,
7,
1,
0,

9,
5,
8,
5,
2,
9,
0,
8,
8,
0,
6,
9,
1,
1,
6,
3,
7,
6,
6,
0,
6,
6,
1,
7,
1,
5,
8,
3,
6,
6,
8,
6,
8,
4,
6,
6,
1,
3,
8,
3,
4,
1,
7,

1,
3,
8,
5,
1,
1,
4,
0,
9,
3,
7,
4,
9,
9,
2,
4,
9,
9,
1,
0,
5,
9,
0,
8,
2,
1,
2,
0,
5,
6,
3,
2,
7,
8,
8,
6,
0,
7,
9,
4,
5,
6,
4,

2,
1,
1,
2,
1,
5,
9,
9,
0,
8,
4,
1,
1,
6,
3,
3,
9,
0,
7,
9,
7,
7,
9,
1,
5,
1,
6,
6,
8,
7,
1,
3,
0,
3,
3,
2,
4,
5,
7,
5,
9,
0,
3,

4,
0,
4,
4,
6,
0,
0,
6,
6,
0,
8,
1,
6,
2,
9,
2,
5,
9,
6,
7,
4,
1,
8,
7,
3,
6,
9,
3,
0,
4,
0,
5,
1,
0,
3,
4,
8,
5,
4,
7,
2,
3,
9,

7,
6,
7,
1,
4,
7,
0,
1,
7,
3,
1,
8,
4,
4,
2,
0,
2,
2,
0,
0,
9,
0,
9,
6,
8,
2,
7,
7,
4,
0,
3,
0,
8,
9,
4,
2,
7,
2,
5,
2,
5,
1,
9,

4,
8,
5,
1,
7,
4,
4,
0,
6,
9,
0,
7,
8,
8,
9,
9,
3,
3,
4,
0,
4,
5,
6,
6,
0,
1,
0,
8,
0,
4,
8,
8,
8,
1,
5,
2,
6,
8,
1,
0,
0,
7,
7,
5,

9,
6,
2,
8,
3,
4,
7,
3,
9,
0,
1,
2,
4,
8,
1,
8,
6,
4,
4,
5,
7,
1,
3,
9,
8,
0,
1,
7,
5,
8,
2,
8,
0,
4,
1,
8,
9,
8,
2,
9,
9,
2,
7,

5,
7,
3,
8,
8,
4,
4,
2,
7,
1,
6,
4,
0,
4,
6,
9,
7,
6,
2,
5,
5,
1,
7,
2,
2,
2,
9,
5,
4,
2,
7,
8,
1,
3,
4,
3,
7,
6,
9,
8,
0,
6,
0,

2,
2,
2,
1,
8,
4,
0,
1,
8,
8,
1,
5,
7,
6,
4,
5,
8,
7,
1,
9,
1,
9,
8,
4,
7,
3,
8,
8,
2,
6,
6,
7,
1,
6,
8,
1,
9,
7,
8,
3,
0,
1,
0,

8,
8,
3,
0,
0,
1,
5,
0,
8,
8,
7,
9,
9,
0,
9,
4,
1,
3,
6,
6,
4,
4,
7,
5,
6,
0,
8,
0,
3,
2,
8,
4,
6,
9,
9,
7,
0,
3,
3,
6,
7,
4,
9,

1,
6,
2,
7,
2,
2,
0,
6,
7,
5,
7,
6,
8,
9,
0,
9,
4,
4,
7,
0,
9,
4,
9,
6,
9,
4,
5,
5,
7,
9,
2,
4,
5,
1,
4,
3,
9,
6,
5,
6,
9,
3,
3,
5,

0,
7,
2,
1,
3,
6,
4,
0,
0,
2,
5,
0,
1,
0,
2,
3,
9,
8,
4,
9,
8,
0,
2,
6,
4,
4,
0,
1,
8,
8,
3,
6,
9,
6,
6,
7,
8,
2,
4,
5,
7,
6,
5,

3,
0,
5,
0,
5,
0,
8,
2,
6,
7,
3,
8,
2,
1,
7,
6,
7,
1,
0,
9,
5,
5,
0,
1,
7,
6,
9,
0,
4,
7,
7,
1,
5,
9,
4,
0,
8,
5,
9,
9,
6,
7,
1,

8,
3,
2,
3,
8,
2,
2,
4,
6,
0,
0,
5,
3,
8,
2,
3,
7,
2,
9,
3,
8,
7,
8,
2,
7,
9,
0,
2,
3,
2,
2,
2,
3,
3,
6,
2,
3,
2,
2,
8,
0,
5,
5,
1,

4,
5,
6,
6,
2,
7,
0,
1,
7,
7,
8,
2,
9,
2,
2,
4,
2,
1,
1,
1,
6,
6,
6,
5,
1,
1,
7,
0,
4,
3,
3,
7,
1,
2,
3,
5,
5,
5,
6,
1,
4,
3,
7,

8,
8,
3,
6,
6,
2,
3,
0,
9,
4,
3,
8,
0,
0,
1,
1,
5,
4,
9,
3,
1,
8,
9,
3,
9,
9,
2,
9,
4,
8,
2,
9,
8,
8,
1,
5,
3,
6,
8,
7,
6,
9,
8,

0,
6,
4,
0,
0,
2,
5,
8,
2,
0,
2,
7,
6,
9,
7,
1,
5,
5,
6,
6,
3,
6,
2,
4,
7,
0,
5,
6,
4,
6,
5,
2,
4,
6,
1,
6,
0,
4,
0,
3,
1,
8,
5,

4,
4,
1,
7,
3,
9,
4,
7,
9,
7,
3,
7,
2,
8,
4,
6,
6,
1,
2,
9,
0,
4,
8,
7,
3,
9,
8,
7,
7,
0,
2,
4,
1,
1,
4,
1,
5,
4,
0,
5,
6,
2,
8,

```
5,  
0,  
2,  
1,  
3,  
5,  
7,  
3,  
5,  
1,  
3,  
5,  
...],  
b'data': array([[ 59,  43,  50, ..., 140,  84,  72],  
               [154, 126, 105, ..., 139, 142, 144],  
               [255, 253, 253, ...,  83,  83,  84],  
               ...,  
               [ 71,   60,   74, ...,   68,   69,   68],  
               [250, 254, 211, ..., 215, 255, 254],  
               [ 62,   61,   60, ..., 130, 130, 131]], dtype=uint8),  
b'filenames': [b'leptodactylus_pentadactylus_s_000004.png',  
  b'camion_s_000148.png',  
  b'tipper_truck_s_001250.png',  
  b'americian_elk_s_001521.png',  
  b'station_wagon_s_000293.png',  
  b'coupe_s_001735.png',  
  b'cassowary_s_001300.png',  
  b'cow_pony_s_001168.png',  
  b'sea_boat_s_001584.png',  
  b'tabby_s_001355.png',  
  b'muntjac_s_001000.png',  
  b'arabian_s_001354.png',  
  b'quarter_horse_s_000672.png',  
  b'passerine_s_000343.png',  
  b'camion_s_001895.png',  
  b'trailer_truck_s_000335.png',  
  b'dumper_s_000821.png',  
  b'alley_cat_s_000200.png',  
  b'accentor_s_000677.png',  
  b'frog_s_001671.png',  
  b'capreolus_capreolus_s_000051.png',  
  b'tomcat_s_000772.png',  
  b'pickerel_frog_s_000446.png',
```

```
b'bufo_s_001242.png',
b'cassowary_s_001246.png',
b'toad_s_001748.png',
b'cat_s_000081.png',
b'chihuahua_s_000825.png',
b'alces_alces_s_000959.png',
b'stealth_bomber_s_000554.png',
b'twinjet_s_000663.png',
b'trucking_rig_s_001402.png',
b'auto_s_000609.png',
b'tabby_cat_s_000983.png',
b'wapiti_s_000416.png',
b'monoplane_s_000895.png',
b'true_cat_s_000247.png',
b'tennessee_walker_s_000486.png',
b'house_cat_s_000243.png',
b'house_cat_s_001196.png',
b'pekinese_s_001337.png',
b'ostrich_s_001368.png',
b'ostrich_s_001150.png',
b'stallion_s_000046.png',
b'station_waggon_s_000041.png',
b'coupe_s_001944.png',
b'estate_car_s_000580.png',
b'accentor_s_000759.png',
b'emu_novaehollandiae_s_000795.png',
b'dive_bomber_s_001390.png',
b'articulated_lorry_s_000131.png',
b'pekinese_s_001093.png',
b'broodmare_s_001463.png',
b'delivery_truck_s_000834.png',
b'songbird_s_001052.png',
b'emu_s_000692.png',
b'puppy_s_000115.png',
b'wagtail_s_001821.png',
b'dama_dama_s_000658.png',
b'domestic_cat_s_001970.png',
b'ambulance_s_003039.png',
b'convertible_s_001763.png',
b'tank_ship_s_001229.png',
b'cassowary_s_001055.png',
b'wagon_s_001142.png',
b'police_cruiser_s_000620.png',
```

```
b'moose_s_002308.png',
b'aerial_ladder_truck_s_000584.png',
b'saddle_horse_s_000717.png',
b'tanker_s_001350.png',
b'mongrel_s_001571.png',
b'truck_s_000835.png',
b'pickerel_frog_s_001195.png',
b'lipizzan_s_000399.png',
b'tabby_s_000074.png',
b'automobile_s_001887.png',
b'moving_van_s_001665.png',
b'attack_aircraft_s_000153.png',
b'domestic_cat_s_001596.png',
b'compact_car_s_000048.png',
b'domestic_cat_s_000009.png',
b'pekingese_s_002089.png',
b'capreolus_capreolus_s_001095.png',
b'blenheim_spaniel_s_001103.png',
b'stallion_s_000040.png',
b'stud_mare_s_001672.png',
b'elk_s_000920.png',
b'lipizzan_s_001123.png',
b'fire_truck_s_002721.png',
b'elk_s_001888.png',
b'finch_s_000750.png',
b'tabby_s_000880.png',
b'banana_boat_s_001324.png',
b'twinjet_s_000591.png',
b'shooting_brake_s_000029.png',
b'rana_pipiens_s_000101.png',
b'station_wagon_s_001387.png',
b'station_wagon_s_002712.png',
b'odocoileus_hemionus_s_000221.png',
b'convertible_s_001715.png',
b'abandoned_ship_s_000574.png',
b'true_cat_s_000114.png',
b'dustcart_s_000063.png',
b'frog_s_000797.png',
b'green_frog_s_001320.png',
b'compact_car_s_001038.png',
b'freighter_s_001351.png',
b'mutt_s_000251.png',
b'accentor_s_000303.png',
```

```
b'lorry_s_001154.png',
b'fire_truck_s_000894.png',
b'cargo_vessel_s_000731.png',
b'cruiser_s_000774.png',
b'lippizan_s_000359.png',
b'broodmare_s_001741.png',
b'fighter_aircraft_s_000876.png',
b'amphibious_aircraft_s_000216.png',
b'leopard_frog_s_000339.png',
b'trucking_rig_s_001315.png',
b'shooting_brake_s_000886.png',
b'pipit_s_000549.png',
b'ostrich_s_002148.png',
b'trucking_rig_s_001600.png',
b'alauda_arvensis_s_000755.png',
b'rana_temporaria_s_001087.png',
b'bufo_s_000136.png',
b'estate_car_s_000529.png',
b'aerial_ladder_truck_s_000997.png',
b'toy_spaniel_s_000384.png',
b'amphibious_aircraft_s_001195.png',
b'fallow_deer_s_001598.png',
b'quarter_horse_s_000414.png',
b'anuran_s_000712.png',
b'arabian_s_001366.png',
b'auto_s_000040.png',
b'cargo_ship_s_001063.png',
b'motorcar_s_000121.png',
b'motorcar_s_000305.png',
b'anthus_pratensis_s_001071.png',
b'cruiser_s_000294.png',
b'wagon_s_000763.png',
b'alley_cat_s_002579.png',
b'tabby_cat_s_000825.png',
b'spadefoot_s_000051.png',
b'wagtail_s_002075.png',
b'fallow_deer_s_001697.png',
b'garbage_truck_s_000777.png',
b'moving范_s_000067.png',
b'mongrel_s_000696.png',
b'red_deer_s_000741.png',
b'true_cat_s_001768.png',
b'spadefoot_s_000377.png',
```

```
b'lipizzan_s_001233.png',
b'japanese_deer_s_000163.png',
b'toad_s_002436.png',
b'police_boat_s_001421.png',
b'puppy_s_002062.png',
b'lapdog_s_001869.png',
b'elk_s_001196.png',
b'domestic_cat_s_001235.png',
b'car_s_000057.png',
b'lightship_s_000064.png',
b'fallow_deer_s_001997.png',
b'quarter_horse_s_000022.png',
b'bufo_calamita_s_000566.png',
b'biplane_s_000624.png',
b'fire_truck_s_001616.png',
b'japanese_spaniel_s_000037.png',
b'ambulance_s_001394.png',
b'domestic_cat_s_001056.png',
b'tugboat_s_001384.png',
b'emu_s_000072.png',
b'gelding_s_000206.png',
b'puppy_s_000637.png',
b'tabby_cat_s_002607.png',
b'elk_s_001198.png',
b'estate_car_s_001496.png',
b'peke_s_001338.png',
b'arabian_s_001018.png',
b'fighter_aircraft_s_000538.png',
b'fallow_deer_s_001164.png',
b'lippizan_s_000390.png',
b'toy_dog_s_000932.png',
b'maltese_s_002151.png',
b'car_s_000002.png',
b'jumbo_jet_s_000697.png',
b'dump_truck_s_000008.png',
b'spadefoot_s_000084.png',
b'motortruck_s_000014.png',
b'jetliner_s_001502.png',
b'tanker_s_000157.png',
b'riding_horse_s_001870.png',
b'cabin_cruiser_s_000096.png',
b'dredger_s_001623.png',
b'pipit_s_001102.png',
```

```
b'mongrel_s_002430.png',
b'skylark_s_000150.png',
b'cat_s_001604.png',
b'puppy_s_000211.png',
b'airbus_s_001124.png',
b'leopard_frog_s_001021.png',
b'wagon_s_001824.png',
b'garbage_truck_s_000147.png',
b'tabby_s_001354.png',
b'rana_temporaria_s_000295.png',
b'lorry_s_000364.png',
b'estate_car_s_000075.png',
b'tabby_s_000970.png',
b'fire_engine_s_000985.png',
b'toad_s_000022.png',
b'texas_toad_s_000215.png',
b'broodmare_s_000707.png',
b'car_s_000286.png',
b'twinjet_s_000565.png',
b'truck_s_001385.png',
b'toy_dog_s_000752.png',
b'police_boat_s_000092.png',
b'toy_dog_s_001197.png',
b'cassowary_s_002446.png',
b'articulated_lorry_s_000050.png',
b'monoplane_s_000781.png',
b'lightship_s_000343.png',
b'powerboat_s_001486.png',
b'jumbo_jet_s_000071.png',
b'bufo_debilis_s_000024.png',
b'transporter_s_000272.png',
b'estate_car_s_000743.png',
b'motorcar_s_000697.png',
b'rana_catesbeiana_s_001655.png',
b'tabby_s_000675.png',
b'remount_s_000285.png',
b'rana_catesbeiana_s_001521.png',
b'true_frog_s_000223.png',
b'airliner_s_002304.png',
b'natterjack_s_000987.png',
b'leptodactylid_s_000082.png',
b'coupe_s_001892.png',
b'quarter_horse_s_000419.png',
```

```
b'car_s_001323.png',
b'maltese_dog_s_001043.png',
b'boat_s_000543.png',
b'tabby_cat_s_001537.png',
b'rana_catesbeiana_s_001096.png',
b'bufo_bufo_s_000326.png',
b'pilot_boat_s_000646.png',
b'toad_frog_s_001672.png',
b'boat_s_000988.png',
b'deer_s_001383.png',
b'bufo_bufo_s_000261.png',
b'leopard_frog_s_000763.png',
b'coupe_s_002178.png',
b'cat_s_000663.png',
b'ship_s_000175.png',
b'house_cat_s_001471.png',
b'stag_s_001458.png',
b'station_wagon_s_001337.png',
b'quarter_horse_s_002201.png',
b'coupe_s_000698.png',
b'tomcat_s_002553.png',
b'scow_s_001266.png',
b'chihuahua_s_000385.png',
b'convertible_s_001500.png',
b'convertible_s_000408.png',
b'odocoileus_hemionus_s_000512.png',
b'jumbo_jet_s_000971.png',
b'truck_s_000650.png',
b'domestic_cat_s_001288.png',
b>walking_horse_s_000194.png',
b'elk_s_001657.png',
b'tipper_truck_s_000287.png',
b'dumper_s_000648.png',
b'struthio_camelus_s_000225.png',
b'woodland_caribou_s_000803.png',
b'ladder_truck_s_000795.png',
b'pantechnicon_s_000230.png',
b'shooting_brake_s_000229.png',
b'jetliner_s_001349.png',
b'toy_dog_s_001280.png',
b'dump_truck_s_002104.png',
b'twinjet_s_001360.png',
b'tugboat_s_000715.png',
```

```
b'passerine_s_000360.png',
b'automobile_s_002337.png',
b'honey_eater_s_001422.png',
b'attack_aircraft_s_001304.png',
b'toy_dog_s_000114.png',
b'anuran_s_000550.png',
b'mouser_s_000382.png',
b'sparrow_s_000037.png',
b'stud_mare_s_001038.png',
b'speedboat_s_002588.png',
b'passenger_ship_s_002153.png',
b'bufo_bufo_s_001876.png',
b'biplane_s_000651.png',
b'gelding_s_001785.png',
b'trucking_rig_s_001598.png',
b'deer_s_000309.png',
b'pekingese_s_001946.png',
b'frog_s_002802.png',
b'cervus_unicolor_s_000290.png',
b'honey_eater_s_000478.png',
b'automobile_s_001152.png',
b'coupe_s_000846.png',
b'flying_bird_s_000252.png',
b'automobile_s_002343.png',
b'japanese/spaniel_s_001162.png',
b'fire_truck_s_000043.png',
b'semi_s_001207.png',
b'fighter_aircraft_s_001840.png',
b'ship_s_001976.png',
b'caribou_s_001222.png',
b'police_cruiser_s_001347.png',
b'coupe_s_000042.png',
b'bufo_bufo_s_000995.png',
b'tabby_s_001148.png',
b'true_cat_s_000476.png',
b'garbage_truck_s_001442.png',
b'fighter_aircraft_s_000385.png',
b'quarter_horse_s_001602.png',
b'camion_s_000116.png',
b'tennessee_walker_s_000539.png',
b'lipizzan_s_000452.png',
b'trucking_rig_s_001594.png',
b'shooting_brake_s_001611.png',
```

```
b'puppy_s_002096.png',
b'auto_s_000855.png',
b'bufo_marinus_s_001364.png',
b'natterjack_s_000036.png',
b'pontoon_s_000643.png',
b'stallion_s_001256.png',
b'coupe_s_001863.png',
b'house_cat_s_000079.png',
b'jetliner_s_000981.png',
b'felis_catus_s_000853.png',
b'tabby_cat_s_000827.png',
b'struthio_camelus_s_001250.png',
b'fawn_s_001915.png',
b'canis_familiaris_s_001298.png',
b'lipizzan_s_002122.png',
b'toy_spaniel_s_001133.png',
b'dump_truck_s_000029.png',
b'fighter_s_001042.png',
b'domestic_cat_s_000493.png',
b'cervus_elaphus_s_001764.png',
b'airbus_s_000627.png',
b'fawn_s_001452.png',
b'fallow_deer_s_000959.png',
b'spadefoot_s_000777.png',
b'seaplane_s_001639.png',
b'plane_s_001066.png',
b'bufo_viridis_s_000348.png',
b'crapaud_s_000051.png',
b'seaplane_s_001447.png',
b'boat_s_000404.png',
b'convertible_s_001242.png',
b'toad_s_000201.png',
b'finch_s_000494.png',
b'fire_engine_s_001437.png',
b'pipit_s_000226.png',
b'toy_dog_s_000960.png',
b'delivery_truck_s_001121.png',
b'bufo_marinus_s_001506.png',
b'studhorse_s_000010.png',
b'woodland_caribou_s_001003.png',
b'station_wagon_s_000260.png',
b'pilot_boat_s_000356.png',
b'stud_mare_s_001497.png',
```

```
b'tabby_cat_s_001966.png',
b'american_toad_s_000006.png',
b'tipper_s_000010.png',
b'tomcat_s_002230.png',
b'dive_bomber_s_001120.png',
b'wapiti_s_000802.png',
b'biplane_s_000862.png',
b'blenheim_spaniel_s_000589.png',
b'shooting_brake_s_001161.png',
b'jetliner_s_000400.png',
b'tabby_s_000923.png',
b'alces_alces_s_000361.png',
b'police_boat_s_000002.png',
b'mutt_s_001696.png',
b'cervus_elaphus_s_001695.png',
b'quarter_horse_s_000752.png',
b'accentor_s_000449.png',
b'tabby_cat_s_000144.png',
b'fire_engine_s_001513.png',
b'tennessee_walking_horse_s_001187.png',
b'bufo_s_001247.png',
b'arabian_s_002147.png',
b'compact_s_001258.png',
b'elk_s_002339.png',
b'stud_mare_s_001384.png',
b'fighter_aircraft_s_000745.png',
b'police_cruiser_s_001047.png',
b'lipizzan_s_001764.png',
b'cat_s_000331.png',
b'taxi_s_000701.png',
b'boat_s_000432.png',
b'japanese_deer_s_000942.png',
b'sambar_s_000076.png',
b'ostrich_s_001559.png',
b'dive_bomber_s_000185.png',
b'lark_s_001532.png',
b'flightless_bird_s_000480.png',
b'stealth_fighter_s_001312.png',
b'jumbojet_s_000789.png',
b'delivery_truck_s_000510.png',
b'biplane_s_001901.png',
b'lorry_s_001185.png',
b'crapaud_s_000511.png',
```

```
b'boat_s_001280.png',
b'bird_s_001357.png',
b'stud_mare_s_001477.png',
b'arabian_s_000751.png',
b'rangifer_caribou_s_000586.png',
b'monoplane_s_000036.png',
b'house_cat_s_001321.png',
b'stealth_fighter_s_000240.png',
b'oil_tanker_s_000129.png',
b'tipper_truck_s_000416.png',
b'roe_deer_s_000358.png',
b'rhea_americana_s_000475.png',
b'lippizaner_s_000517.png',
b'cassowary_s_000297.png',
b'pekingese_s_001675.png',
b'accentor_s_001058.png',
b'lapdog_s_000227.png',
b'auto_s_001153.png',
b'fire_engine_s_000443.png',
b'muntjac_s_000815.png',
b'lightship_s_000090.png',
b'dog_s_002441.png',
b'auto_s_001951.png',
b'arabian_s_000328.png',
b'odocoileus_hemionus_s_000188.png',
b'barking_deer_s_000019.png',
b'dive_bomber_s_001486.png',
b'bufo_viridis_s_000503.png',
b>wrecker_s_002294.png',
b'monoplane_s_000206.png',
b'arabian_s_002282.png',
b'speedboat_s_002160.png',
b'cargo_vessel_s_002136.png',
b'garbage_truck_s_000726.png',
b'delivery_truck_s_001468.png',
b'house_cat_s_002240.png',
b'house_cat_s_000045.png',
b'caribou_s_002007.png',
b'propeller_plane_s_000451.png',
b'fallow_deer_s_001082.png',
b'dog_s_001897.png',
b'bufo_bufo_s_001631.png',
b'spadefoot_s_000191.png',
```

```
b'multiengine_airplane_s_000186.png',
b'auto_s_002359.png',
b'stealth_bomber_s_001607.png',
b'passenger_ship_s_000582.png',
b'airliner_s_001297.png',
b'fallow_deer_s_000973.png',
b'speedboat_s_000178.png',
b'boat_s_001444.png',
b'estate_car_s_000712.png',
b'english_toy_spaniel_s_000888.png',
b'passerine_s_000699.png',
b'rana_pipiens_s_000286.png',
b'tanker_s_000025.png',
b'car_s_001956.png',
b'stealth_bomber_s_000424.png',
b'airliner_s_001070.png',
b'stud_mare_s_001348.png',
b'quarter_horse_s_000282.png',
b'maltese_s_001818.png',
b'lorry_s_001786.png',
b'true_frog_s_000064.png',
b'cassowary_s_001372.png',
b'houseboat_s_000626.png',
b'domestic_cat_s_000035.png',
b'deer_s_001596.png',
b'cow_pony_s_000803.png',
b'tabby_cat_s_001466.png',
b'tipper_truck_s_001525.png',
b'stealth_fighter_s_000533.png',
b'station_wagon_s_000750.png',
b'wagtail_s_001549.png',
b'rangifer_caribou_s_000985.png',
b'cargo_ship_s_000408.png',
b'cab_s_000788.png',
b'flatboat_s_000044.png',
b'bufo_viridis_s_000457.png',
b'fallow_deer_s_000040.png',
b'barren_ground_caribou_s_000077.png',
b'pekinese_s_000453.png',
b'female_horse_s_000358.png',
b'compact_car_s_000869.png',
b'tabby_cat_s_002454.png',
b'delivery_truck_s_000864.png',
```

```
b'cargo_ship_s_000802.png',
b'attack_aircraft_s_000037.png',
b'convertible_s_000037.png',
b'broodmare_s_001448.png',
b'toy_spaniel_s_001286.png',
b'packet_boat_s_001206.png',
b'ostrich_s_001564.png',
b'hospital_ship_s_001658.png',
b'stealth_bomber_s_001035.png',
b'stag_s_002583.png',
b'police_cruiser_s_000629.png',
b'merchant_ship_s_001242.png',
b'dump_truck_s_001584.png',
b'cargo_ship_s_001830.png',
b'nandu_s_000721.png',
b'garbage_truck_s_001119.png',
b'lorry_s_002317.png',
b'wagtail_s_001445.png',
b'lippizan_s_002049.png',
b'maltese_s_000948.png',
b'quarter_horse_s_000798.png',
b'house_cat_s_001472.png',
b'tank_ship_s_001155.png',
b'hospital_ship_s_001838.png',
b'mule_deer_s_000581.png',
b'deer_s_001329.png',
b'rhea_americana_s_000268.png',
b'lippizan_s_000440.png',
b'convertible_s_002117.png',
b'bufo_marinus_s_001567.png',
b'roe_deer_s_000690.png',
b'twinjet_s_000481.png',
b'amERICAN_elk_s_001743.png',
b'bufo_viridis_s_000254.png',
b'dumper_s_000179.png',
b'lippizan_s_000323.png',
b'leopard_frog_s_000058.png',
b'prunella_modularis_s_000578.png',
b'maltese_s_001867.png',
b'toy_dog_s_001358.png',
b'police_cruiser_s_000500.png',
b'broodmare_s_001589.png',
b'passerine_s_000381.png',
```

```
b'accentor_s_000424.png',
b'finch_s_000617.png',
b'trucking_rig_s_001111.png',
b'canis_familiaris_s_001059.png',
b'cervus_elaphus_s_000813.png',
b'cassowary_s_000339.png',
b'buckskin_s_000325.png',
b'lightship_s_001019.png',
b'shooting_brake_s_001330.png',
b'tomcat_s_002649.png',
b'barren_ground_caribou_s_000174.png',
b'cat_s_000572.png',
b>walking_horse_s_002079.png',
b'rana_temporaria_s_001139.png',
b'tipper_s_000007.png',
b'freighter_s_000661.png',
b'stealth_fighter_s_000020.png',
b'goliath_frog_s_000205.png',
b'stealth_bomber_s_000703.png',
b'prunella_modularis_s_000795.png',
b'struthio_camelus_s_000331.png',
b'dunnock_s_000956.png',
b'auto_s_001284.png',
b'scow_s_000831.png',
b>fallow_deer_s_001256.png',
b'stealth_fighter_s_001297.png',
b'estate_car_s_000845.png',
b'ship_s_001691.png',
b'abandoned_ship_s_000977.png',
b>wagon_s_001298.png',
b'feist_s_000354.png',
b'lipizzan_s_001600.png',
b'leopard_frog_s_001700.png',
b'cervus_elaphus_s_000066.png',
b'toy_s_000517.png',
b'cargo_ship_s_000556.png',
b'stallion_s_001717.png',
b'coupe_s_001004.png',
b'camion_s_001068.png',
b'coupe_s_000001.png',
b'garbage_truck_s_001253.png',
b'sea_boat_s_001960.png',
b'capreolus_capreolus_s_001169.png',
```

```
b'lippizan_s_000681.png',
b'tabby_s_000512.png',
b'pontoon_s_000271.png',
b'sea_boat_s_001502.png',
b'lark_s_000700.png',
b'leopard_frog_s_001074.png',
b'rana_palustris_s_000022.png',
b'lippizaner_s_001181.png',
b'wagon_s_000155.png',
b'americana_toad_s_001410.png',
b'cargo_vessel_s_000659.png',
b'station_wagon_s_001705.png',
b'transporter_s_000076.png',
b'broodmare_s_001098.png',
b'passenger_ship_s_000050.png',
b'tabby_cat_s_000484.png',
b'fighter_s_000276.png',
b'car_s_000506.png',
b'stealth_fighter_s_000139.png',
b'pleasure_craft_s_000076.png',
b'cruiser_s_000744.png',
b'tabby_cat_s_002478.png',
b'attack_aircraft_s_000423.png',
b'stealth_fighter_s_000460.png',
b'police_cruiser_s_000787.png',
b'mongrel_s_000381.png',
b'attack_aircraft_s_001166.png',
b'supertanker_s_000557.png',
b'tugboat_s_000807.png',
b'stud_mare_s_000413.png',
b'lorry_s_001039.png',
b'trailer_truck_s_001496.png',
b'airliner_s_001477.png',
b'camion_s_001521.png',
b'alces_alces_s_000404.png',
b'automobile_s_000315.png',
b'tabby_cat_s_001542.png',
b'pickerel_frog_s_000003.png',
b'spring_frog_s_000318.png',
b'roebuck_s_000153.png',
b'reindeer_s_000121.png',
b'male_horse_s_000008.png',
b'dog_s_000052.png',
```

```
b'bufo_bufo_s_001959.png',
b'stealth_bomber_s_000492.png',
b'pilot_boat_s_001819.png',
b'attack_aircraft_s_001187.png',
b'house_cat_s_000812.png',
b'cassowary_s_000138.png',
b'boat_s_000490.png',
b'wapiti_s_000884.png',
b'bufo_bufo_s_001492.png',
b'delivery_truck_s_000157.png',
b'truck_s_000060.png',
b'stud_mare_s_000506.png',
b'fighter_aircraft_s_000655.png',
b>true_cat_s_000127.png',
b'tabby_s_000228.png',
b'americangreen_toad_s_000007.png',
b'quarter_horse_s_001833.png',
b'odocoileus_hemionus_s_001090.png',
b'tipper_lorry_s_000527.png',
b'compact_car_s_001200.png',
b'pickerel_frog_s_000242.png',
b'finch_s_000308.png',
b'arabian_s_002456.png',
b'dunnock_s_001476.png',
b'emu_s_002196.png',
b'plane_s_000065.png',
b'european_toad_s_000360.png',
b'arabian_s_002383.png',
b'dog_s_002098.png',
b'quarter_horse_s_001757.png',
b'rana_temporaria_s_001922.png',
b'cargo_ship_s_001318.png',
b'tip_truck_s_001017.png',
b'biplane_s_001055.png',
b'camion_s_001255.png',
b'red_deer_s_000756.png',
b'capreolus_capreolus_s_000822.png',
b'lippizaner_s_000840.png',
b'propeller_plane_s_001438.png',
b'dumper_s_000646.png',
b'capreolus_capreolus_s_000284.png',
b'truck_s_000526.png',
b'bufo_s_001473.png',
```

```
b'fire_engine_s_001449.png',
b'elk_s_001164.png',
b'canis_familiaris_s_001319.png',
b'tennessee_walking_horse_s_000005.png',
b'truck_s_000838.png',
b'rhea_s_001649.png',
b'mule_deer_s_002514.png',
b'canis_familiaris_s_000143.png',
b'police_cruiser_s_001073.png',
b'cervus_sika_s_000273.png',
b'cat_s_001584.png',
b'tipper_lorry_s_000313.png',
b'bufo_americanus_s_001202.png',
b'english_toy_spaniel_s_000735.png',
b'bufo_calamita_s_000043.png',
b'tipper_s_000180.png',
b'tabby_cat_s_000881.png',
b'tabby_cat_s_001644.png',
b'blenheim_spaniel_s_000330.png',
b'attack_aircraft_s_001138.png',
b'arabian_s_002270.png',
b'cassowary_s_000845.png',
b'wagon_s_000692.png',
b'tabby_cat_s_000319.png',
b'americana_toad_s_000077.png',
b'moose_s_001500.png',
b'airliner_s_001406.png',
b'fighter_aircraft_s_000966.png',
b'dunnock_s_000055.png',
b'pekingese_s_002399.png',
b'twinjet_s_000739.png',
b'automobile_s_002249.png',
b'airliner_s_000669.png',
b'wagtail_s_001656.png',
b'tabby_s_001949.png',
b'dump_truck_s_001009.png',
b'freighter_s_000436.png',
b'fallow_deer_s_001171.png',
b'trucking_rig_s_000684.png',
b'ship_s_000249.png',
b'monoplane_s_000416.png',
b'songbird_s_002066.png',
b'rana_catesbeiana_s_000602.png',
```

```
b'red_deer_s_000536.png',
b'elk_s_001583.png',
b'stealth_fighter_s_001313.png',
b'station_wagon_s_002498.png',
b'pleasure_boat_s_001949.png',
b'cargo_ship_s_000060.png',
b'tabby_cat_s_001767.png',
b'craaud_s_000055.png',
b'dump_truck_s_002008.png',
b'bufo_viridis_s_000506.png',
b'bullfrog_s_001295.png',
b'riding_horse_s_001884.png',
b'boat_s_002344.png',
b'ostrich_s_000950.png',
b'muntjac_s_000681.png',
b'pekingese_s_000279.png',
b'lipizzan_s_002052.png',
b'rana_pipiens_s_000570.png',
b'toy_s_002250.png',
b'tabby_s_001940.png',
b'dive_bomber_s_001294.png',
b'mutt_s_000086.png',
b'biplane_s_000701.png',
b'toy_s_000964.png',
b'jetliner_s_000807.png',
b'barge_s_000426.png',
b'cassowary_s_000255.png',
b'rana_clamitans_s_000059.png',
b'quarter_horse_s_000222.png',
b'tomcat_s_000377.png',
b'scow_s_000687.png',
b'emu_s_001571.png',
b'coupe_s_001625.png',
b'roan_s_000199.png',
b'bufo_bufo_s_000626.png',
b'tennessee_walking_horse_s_001470.png',
b'station_wagon_s_001497.png',
b'jumbo_jet_s_001056.png',
b'trailer_truck_s_000099.png',
b'puppy_s_000502.png',
b'pekingese_s_002101.png',
b'propeller_plane_s_001193.png',
b'car_s_001009.png',
```

```
b'lipizzan_s_002011.png',
b'green_frog_s_001375.png',
b'tipper_lorry_s_000586.png',
b'jetliner_s_000509.png',
b'musk_deer_s_000211.png',
b'appaloosa_s_002251.png',
b'arabian_s_001642.png',
b'compact_car_s_000435.png',
b'chihuahua_s_000063.png',
b'semi_s_000462.png',
b'wapiti_s_000455.png',
b'twinjet_s_001120.png',
b'cargo_vessel_s_001235.png',
b'peke_s_000274.png',
b'dump_truck_s_001422.png',
b'dump_truck_s_001863.png',
b'tailed_frog_s_000638.png',
b'gelding_s_000965.png',
b'coupe_s_000056.png',
b'pleasure_craft_s_000625.png',
b'house_cat_s_001328.png',
b'struthio_camelus_s_000081.png',
b'domestic_cat_s_000051.png',
b'passenger_ship_s_000948.png',
b'sparrow_s_002697.png',
b'prunella_modularis_s_000302.png',
b'fallow_deer_s_000192.png',
b'rana_palustris_s_000107.png',
b'attack_aircraft_s_001075.png',
b'twinjet_s_000046.png',
b'chihuahua_s_001038.png',
b'tabby_cat_s_001004.png',
b'cargo_vessel_s_001812.png',
b'dunnock_s_000037.png',
b'tabby_s_001740.png',
b'gelding_s_001638.png',
b'anthonus_pratensis_s_001166.png',
b'camion_s_001988.png',
b'alley_cat_s_001645.png',
b'oil_tanker_s_001626.png',
b'tennessee_walking_horse_s_000355.png',
b'containership_s_000812.png',
b'passerine_s_000801.png',
```

```
b'lipizzan_s_000635.png',
b'truck_s_000635.png',
b'twinjet_s_000450.png',
b'honey_eater_s_000702.png',
b'tabby_s_000039.png',
b'pipit_s_000827.png',
b'elephant_bird_s_001004.png',
b'night_bird_s_000241.png',
b'alley_cat_s_001301.png',
b'house_cat_s_000332.png',
b'bufo_americanus_s_000492.png',
b'wagtail_s_001443.png',
b'tabby_cat_s_000587.png',
b'gamecock_s_000098.png',
b'cargo_ship_s_001867.png',
b'seaplane_s_000416.png',
b'english_toy_spaniel_s_001058.png',
b'japanese/spaniel_s_001324.png',
b'coupe_s_000179.png',
b'amERICAN_elk_s_000201.png',
b'chihuahua_s_000666.png',
b'spadefoot_s_000219.png',
b'bufo_s_001733.png',
b'struthio_camelus_s_001368.png',
b'arab_s_000456.png',
b'monoplane_s_001154.png',
b'coupe_s_001690.png',
b'dawn_horse_s_000752.png',
b'tennessee_walking_horse_s_000013.png',
b'cabin_cruiser_s_001500.png',
b'sparrow_s_001750.png',
b'aerial_ladder_truck_s_001306.png',
b'dromaius_novaehollandiae_s_000433.png',
b'passerine_s_000376.png',
b'caribou_s_001447.png',
b'bird_s_000356.png',
b'motorcar_s_001389.png',
b'auto_s_000675.png',
b'automobile_s_001833.png',
b'rana_temporaria_s_000008.png',
b'spring_frog_s_001263.png',
b'barking_frog_s_000397.png',
b'puppy_s_000785.png',
```

```
b'convertible_s_001861.png',
b'convertible_s_000859.png',
b'amERICAN_sADDLE_HORSE_s_000039.png',
b'biPLANE_s_000244.png',
b'red_deer_s_000704.png',
b'tabby_s_001071.png',
b'tabby_cat_s_002548.png',
b'tennessee_wALKER_s_001791.png',
b'auto_s_002048.png',
b'dromaius_novaehollandiae_s_000296.png',
b'felis_catus_s_001460.png',
b'peke_s_000440.png',
b'chihuahua_s_000349.png',
b'lapdog_s_002195.png',
b'bufo_marinus_s_001243.png',
b'convertible_s_001433.png',
b'mule_deer_s_000628.png',
b'tabby_s_000410.png',
b'stud_s_000099.png',
b'passenger_ship_s_001530.png',
b'scow_s_000131.png',
b'tabby_s_001868.png',
b'bufo_bufo_s_002334.png',
b'bufo_viridis_s_000152.png',
b'passerine_s_001306.png',
b'house_cat_s_002116.png',
b'jetliner_s_001188.png',
b'moving_van_s_001359.png',
b'wapiti_s_000663.png',
b'felis_catus_s_000998.png',
b'sea_boat_s_001979.png',
b'fighter_aircraft_s_001776.png',
b'fighter_aircraft_s_001096.png',
b'shooting_brake_s_000774.png',
b'automobile_s_001310.png',
b'pekingese_s_001023.png',
b'amERICAN_elk_s_000455.png',
b'trucking_rig_s_001243.png',
b'house_cat_s_000877.png',
b'taxi_s_000040.png',
b'pilot_boat_s_001100.png',
b'articulated_lorry_s_000223.png',
b'felis_domesticus_s_000164.png',
```

```
b'trucking_rig_s_001300.png',
b'tractor_trailer_s_000254.png',
b'flightless_bird_s_000142.png',
b'dump_truck_s_000939.png',
b'deer_s_000967.png',
b'ship_s_001904.png',
b'wagtail_s_000309.png',
b'trailer_truck_s_000815.png',
b'container_ship_s_002477.png',
b'guard_boat_s_000285.png',
b'estate_car_s_000446.png',
b'puppy_s_001621.png',
b'tabby_cat_s_000689.png',
b'americana_toad_s_000650.png',
b'boat_s_002381.png',
b'female_horse_s_000089.png',
b'bufo_viridis_s_001370.png',
b'ladder_truck_s_000044.png',
b'cargo_ship_s_001321.png',
b'twinjet_s_000547.png',
b'cascades_frog_s_000061.png',
b'odocoileus_hemionus_s_001017.png',
b'attack_aircraft_s_001254.png',
b'airplane_s_00006.png',
b'songbird_s_001232.png',
b'peke_s_000388.png',
b'boat_s_002143.png',
b'accendor_s_001182.png',
b'dive_bomber_s_001066.png',
b'dunnock_s_001650.png',
b'dawn_horse_s_001127.png',
b'european_toad_s_000359.png',
b'wrecker_s_002325.png',
b'lipizzan_s_000118.png',
b'automobile_s_000327.png',
b'pekinese_s_002027.png',
b'maltese_s_001913.png',
b'toad_frog_s_000836.png',
b'leopard_frog_s_001231.png',
b'domestic_cat_s_001066.png',
b'barking_frog_s_000840.png',
b'finch_s_000038.png',
b'capreolus_capreolus_s_000257.png',
```

```
b'lippizan_s_000126.png',
b'attack_aircraft_s_000089.png',
b'mutt_s_002278.png',
b'true_toad_s_000021.png',
b'capreolus_capreolus_s_000178.png',
b'bufo_viridis_s_000600.png',
b'chihuahua_s_000097.png',
b'meadow_pipit_s_001282.png',
b'red_deer_s_000008.png',
b'leopard_frog_s_001528.png',
b'estate_car_s_001220.png',
b'bullfrog_s_000408.png',
b'stealth_fighter_s_001637.png',
b'capreolus_capreolus_s_001669.png',
b'airbus_s_000736.png',
b'tabby_s_000292.png',
b'station_wagon_s_002793.png',
b'boat_s_000765.png',
b'english_toy_spaniel_s_000814.png',
b'european_elk_s_000265.png',
b'roe_deer_s_000371.png',
b'convertible_s_000414.png',
b'cow_pony_s_000145.png',
b'tabby_s_000126.png',
b'trucking_rig_s_001628.png',
b'muntjac_s_001738.png',
b'broodmare_s_000582.png',
b'fire_engine_s_001540.png',
b'gelding_s_000640.png',
b'tabby_cat_s_002533.png',
b'appaloosa_s_000342.png',
b'pipit_s_001066.png',
b'speedboat_s_000410.png',
b'roe_deer_s_001014.png',
b'americana_toad_s_001815.png',
b'bufo_calamita_s_000773.png',
b'estate_car_s_000044.png',
b'cassowary_s_000021.png',
b'lorry_s_001709.png',
b'amphibious_aircraft_s_000141.png',
b'capreolus_capreolus_s_000073.png',
b'motorboat_s_000773.png',
b'lipizzan_s_001658.png',
```

```
b'tabby_cat_s_000531.png',
b'truck_s_000289.png',
b'speedboat_s_001711.png',
b'quarter_horse_s_000557.png',
b'quarter_horse_s_000942.png',
b'jumbo_jet_s_001476.png',
b'struthio_camelus_s_001197.png',
b'fallow_deer_s_001623.png',
b'compact_car_s_000850.png',
b'coupe_s_001912.png',
b'roe_deer_s_000678.png',
b'car_s_000674.png',
b'chihuahua_s_001937.png',
b'red_deer_s_002698.png',
b'jumbojet_s_001587.png',
b'toy_spaniel_s_001692.png',
b'rana_clamitans_s_000766.png',
b'pipit_s_000025.png',
b'cargo_vessel_s_001809.png',
b'pekingese_s_001613.png',
b'dive_bomber_s_000707.png',
b'lark_s_000550.png',
b'estate_car_s_000292.png',
b'tabby_cat_s_001853.png',
b'lapdog_s_001963.png',
b'palfrey_s_000300.png',
b'cat_s_002298.png',
b'mutt_s_000188.png',
b'police_cruiser_s_000471.png',
b'true_cat_s_001598.png',
b'peke_s_000638.png',
...]}
```

4. Display images using matplotlib

```
In [10]: import matplotlib.pyplot as plt
%matplotlib inline

import numpy as np
```

Reshape the image from 3 x 32 x 32 to 32 x 32 x 3

```
In [11]: X = data_batch1[b"data"]
X = X.reshape(10000, 3, 32, 32).transpose(0, 2, 3, 1).astype("uint8") # IMAGE = 32 * 32 & R G B = 3
```

```
In [12]: X[0].max()
```

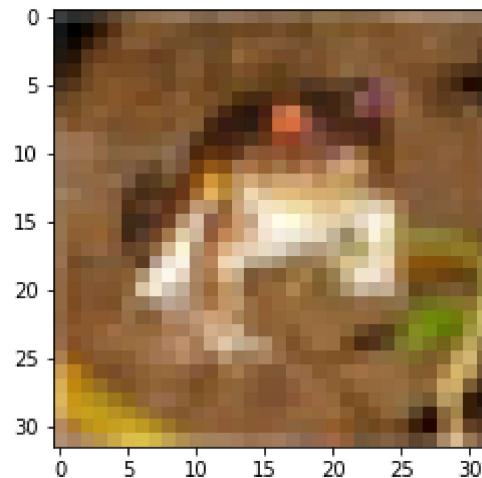
```
Out[12]: 255
```

```
In [13]: (X[0]/255).max()
```

```
Out[13]: 1.0
```

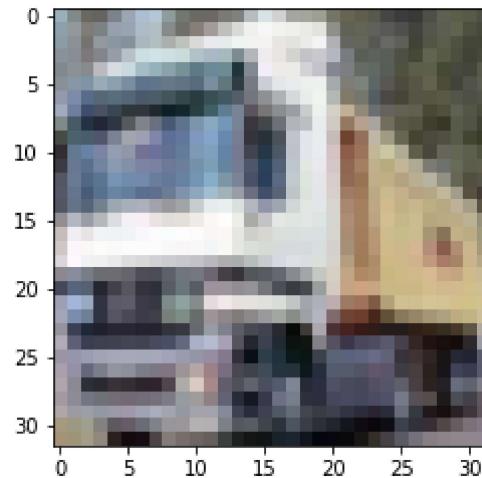
```
In [14]: plt.imshow(X[0])
```

```
Out[14]: <matplotlib.image.AxesImage at 0x2bb37d78978>
```



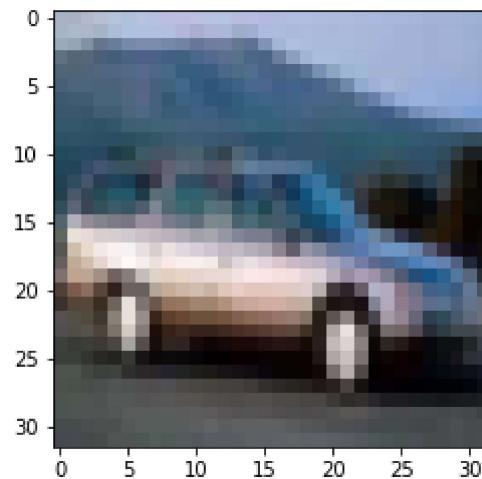
```
In [15]: plt.imshow(X[1])
```

```
Out[15]: <matplotlib.image.AxesImage at 0x2bb37e05eb8>
```



```
In [16]: plt.imshow(X[4])
```

```
Out[16]: <matplotlib.image.AxesImage at 0x2bb37e6f198>
```



5. Helper function to handle data

```
In [17]: def one_hot_encode(vec, vals = 10) :  
    '''  
        For use to one-hot encode the 10 - possible labels.  
    '''  
  
    n = len(vec)  
    out = np.zeros((n, vals))  
    out[range(n), vec] = 1  
    return out
```

Initialize the Training Images, Labels & Test images, Test Labels

```
In [18]: class CifarHelper() :  
  
    def __init__(self) :  
        self.i = 0  
  
        self.all_train_batches = [ data_batch1, data_batch2, data_batch3, data_batch4, data_batch5] # Train data  
        self.test_batch = [test_batch] # Test data  
  
        self.training_images = None  
        self.training_labels = None  
  
        self.test_images = None  
        self.test_labels = None  
  
    def set_up_images(self) :  
  
        print("Setting up Training Images and Labels :")  
  
        self.training_images = np.vstack([d[b"data"] for d in self.all_train_batches])  
        train_len = len(self.training_images)  
        print("Training images count :", train_len)  
  
        self.training_images = self.training_images.reshape(train_len, 3, 32, 32).transpose(0, 2, 3, 1)/255  
        self.training_labels = one_hot_encode(np.hstack([d[b"labels"] for d in self.all_train_batches]), 10)  
        print("\nSetting up Test Images and Labels")  
  
        self.test_images = np.vstack([d[b"data"] for d in self.test_batch])  
        test_len = len(self.test_images)  
        print("Training images count :", test_len)  
  
        self.test_images = self.test_images.reshape(test_len, 3, 32, 32).transpose(0, 2, 3, 1)/255  
        self.test_labels = one_hot_encode(np.hstack([d[b"labels"] for d in self.test_batch]), 10)  
  
    def next_batch(self, batch_size) :  
        x = self.training_images[self.i : self.i + batch_size].reshape(100, 32, 32, 3)  
        y = self.training_labels[self.i : self.i + batch_size]  
  
        self.i = (self.i + batch_size) % len(self.training_images)  
        return x, y # x = image and y = label
```

6. Run the following code to initialize CifaHelper and Set up Images

```
In [19]: ch = CifarHelper()  
ch.set_up_images()
```

```
Setting up Training Images and Labels :  
Training images count : 50000
```

```
Setting up Test Images and Labels  
Training images count : 10000
```

7. Creating the Model

```
In [20]: import tensorflow as tf
```

```
C:\Users\Sreekanth\Anaconda3\lib\site-packages\h5py\__init__.py:36: FutureWarning: Conversion of the second argument of issubdtype from `float` to `np.floating` is deprecated. In future, it will be treated as `np.float64 == np.dtype(float).type`.
```

```
    from ._conv import register_converters as _register_converters
```

```
In [21]: x = tf.placeholder(tf.float32, shape = [None, 32, 32, 3])  
y_true = tf.placeholder(tf.float32, shape = [None, 10] )
```

```
hold_prob = tf.placeholder(tf.float32) # drop outs.
```

8 . Applying the CNN Helper functions. We can get from MNIST

```
In [22]: def init_weights(shape) :
    init_random_dist = tf.truncated_normal(shape, stddev = 0.1)
    return tf.Variable(init_random_dist) # random distribution

def init_bias(shape):
    init_bias_vals = tf.constant(0.1, shape = shape)
    return tf.Variable(init_bias_vals)

def conv2d(x, W):
    return tf.nn.conv2d(x, W, strides =[1,1,1,1], padding ='SAME') # stride = small set of data.

def max_pool_2by2(x):
    return tf.nn.max_pool(x, ksize = [1, 2, 2, 1], strides =[1, 2,2,1], padding ='SAME')

def convolutional_layer(input_x, shape) :
    W = init_weights(shape)
    b = init_bias([shape[3]])
    return tf.nn.relu(conv2d(input_x, W) + b)

def normal_full_layer(input_layer, size) :
    input_size = int(input_layer.get_shape()[1])
    W = init_weights([input_size, size])
    b = init_bias([size])
    return tf.matmul(input_layer, W) + b
```

9 . Create the Layers

```
In [23]: convo_1 = convolutional_layer(x, shape =[4, 4, 3, 32]) # 4 * 4 is filter size. 3 different channels 32 image
size
convo_1_pooling = max_pool_2by2(convo_1)

convo_2 = convolutional_layer(convo_1_pooling, shape =[4, 4, 32, 64]) #
convo_2_pooling = max_pool_2by2(convo_2)
```

10 . Create the flattened layer by reshaping the pooling layer

```
In [24]: 8*8*64
```

```
Out[24]: 4096
```

```
In [25]: convo_2_flat = tf.reshape(convolutional_2_pooling, [-1, 8 * 8 * 64])
```

11. Create the fully connected layer

```
In [26]: full_layer_one = tf.nn.relu(normal_full_layer(convolutional_2_flat, 1024))
```

```
full_one_dropout = tf.nn.dropout(full_layer_one, keep_prob = hold_prob)
```

12. Set output to y_pred

```
In [27]: y_pred = normal_full_layer(full_one_dropout, 10)
```

```
In [28]: y_pred
```

```
Out[28]: <tf.Tensor 'add_3:0' shape=(?, 10) dtype=float32>
```

13 . Apply the Loss function

```
In [29]: cross_entropy = tf.reduce_mean(tf.nn.softmax_cross_entropy_with_logits_v2(labels = y_true, logits = y_pred ))
```

14 . Create the Optimizer

```
In [30]: optimizer = tf.train.AdamOptimizer(learning_rate = 0.0001)
train = optimizer.minimize(cross_entropy)
```

15. Create a variable to initialize all the global tf variables

```
In [31]: init = tf.global_variables_initializer()
```

16. Run the model by creating a graph Session

```
In [32]: with tf.Session() as sess:  
    sess.run(init)  
  
    for i in range(5000) :  
        batch = ch.next_batch(100)  
        sess.run(train, feed_dict = { x: batch[0], y_true: batch[1], hold_prob : 0.6})  
  
        # print out message every 100 steps  
        if (i % 100 == 0) :  
            print('Currently on step {}'.format(i))  
            print('Accuracy is: ')  
  
        # Test the train model  
        matches = tf.equal(tf.argmax(y_pred, 1), tf.argmax(y_true, 1))  
        acc = tf.reduce_mean(tf.cast(matches, tf.float32))  
        print(sess.run(acc, feed_dict = { x: ch.test_images, y_true: ch.test_labels, hold_prob : 1.0}))  
        print('\n')
```

Currently on step 0

Accuracy is:

0.0932

Currently on step 100

Accuracy is:

0.3104

Currently on step 200

Accuracy is:

0.3905

Currently on step 300

Accuracy is:

0.4199

Currently on step 400

Accuracy is:

0.4379

Currently on step 500

Accuracy is:

0.4541

Currently on step 600

Accuracy is:

0.4723

Currently on step 700

Accuracy is:

0.4821

Currently on step 800

Accuracy is:

0.4913

Currently on step 900
Accuracy is:
0.4981

Currently on step 1000
Accuracy is:
0.5115

Currently on step 1100
Accuracy is:
0.5138

Currently on step 1200
Accuracy is:
0.5295

Currently on step 1300
Accuracy is:
0.5309

Currently on step 1400
Accuracy is:
0.541

Currently on step 1500
Accuracy is:
0.5473

Currently on step 1600
Accuracy is:
0.55

Currently on step 1700

Accuracy is:

0.5581

Currently on step 1800

Accuracy is:

0.5628

Currently on step 1900

Accuracy is:

0.569

Currently on step 2000

Accuracy is:

0.5666

Currently on step 2100

Accuracy is:

0.5726

Currently on step 2200

Accuracy is:

0.5815

Currently on step 2300

Accuracy is:

0.5846

Currently on step 2400

Accuracy is:

0.5894

Currently on step 2500

Accuracy is:

0.5874

Currently on step 2600
Accuracy is:
0.5912

Currently on step 2700
Accuracy is:
0.6023

Currently on step 2800
Accuracy is:
0.6014

Currently on step 2900
Accuracy is:
0.6028

Currently on step 3000
Accuracy is:
0.6026

Currently on step 3100
Accuracy is:
0.6089

Currently on step 3200
Accuracy is:
0.6182

Currently on step 3300
Accuracy is:
0.6177

Currently on step 3400
Accuracy is:

0.6155

Currently on step 3500

Accuracy is:

0.6186

Currently on step 3600

Accuracy is:

0.6256

Currently on step 3700

Accuracy is:

0.6242

Currently on step 3800

Accuracy is:

0.622

Currently on step 3900

Accuracy is:

0.6295

Currently on step 4000

Accuracy is:

0.6242

Currently on step 4100

Accuracy is:

0.6332

Currently on step 4200

Accuracy is:

0.6346

Currently on step 4300
Accuracy is:
0.6344

Currently on step 4400
Accuracy is:
0.641

Currently on step 4500
Accuracy is:
0.6368

Currently on step 4600
Accuracy is:
0.644

Currently on step 4700
Accuracy is:
0.6466

Currently on step 4800
Accuracy is:
0.6462

Currently on step 4900
Accuracy is:
0.6506

In []: