ENPM673 - Perception for Autonomous Robots

Project 6

Prateek Arora, Abhinav Modi, Samer Charifa

Due date: 12^{th} *May 2020, 11:59 p.m.*

Submission Guidelines

- The homework is to be completed in group and there should be a single submission per group.
- Your submission on ELMS/Canvas must be a zip file, following the naming convention **YourDirectoryID_proj_6.zip**. If you email ID is abc@umd.edu or abc@terpmail.umd.edu, then your DirectoryID is **abc**. Remember, this is your directory ID and **NOT** your UID. Please provide detailed instructions on how to run your code in README.md file.
- For each section of the homework, explain briefly what you did, and describe any interesting problems you encountered and/or solutions you implemented. Your report should preferable be typeset in LaTeX.
- Please submit only the python script(s) you used to compute the results, the PDF report you generate for the project and a detailed README file (refer to the directory structure below).
- Please don't include the dataset provided to you and your video results in your submission, rather provide the link to your video output in the report.
- Disallowed function: For this particular project you are free to use any in-built and/or third-party functions.

The file tree of your submission SHOULD resemble this:

```
YourDirectoryID_p6.zip
├── Code
│   ├── .py files
│   └── any subdirectories that you may have
├── Report.pdf
└── README.md
```

Introduction

In this project your aim is to classify whether images, in the given dataset, contain either a dog or a cat. This is easy for humans, dogs, and cats. Your computer will find it a bit more difficult.

Data

The given training archive contains 25,000 images of dogs and cats. Train your network on these images to predict the labels for test1.zip (1 = dog, 0 = cat). The testing dataset contains 12500 images. You don't necessarily need to train on all 25000 images.
Download the dataset here

Figure 1: Caption

Resources

In case you don't have access to a machine with a GPU for training the neural network, you can follow this link to use google colab and use GPUs for this project.

Also here you can find a tutorial for using pytorch with google Colab.

Suggested Pipeline

You are required to implement a Convolution Neural Network(CNN) to perform classification on the given dataset. For such classification task, generally VGG-16 architecture is deployed. Your implementation the same using tensorflow, pytorch or keras.
If you are super new to machine learning and deep learning, there are a lot of resources online to learn how to program a simple neural network, tune hyperparameters for CIFAR-10. A good starting point for tensorflow tutorials is the official Tensorflow tutorial and this great tutorial by Hvass Labs.

Improving Accuracy of your neural network

Now that we have a baseline neural network working, let's try to improve the accuracy by doing simple tricks.

1. Standardize your data input if you haven't already. There are a lot of ways to do this. Feel free to search for different methods. A simple way is to scale data from [0,255] to [-1,1].

2. Decay your learning rate as you train or increase your batch size as you train.

3. Augment your data to artificially make your dataset larger.
4. Add Batch Normalization between layers.
5. Change the hyperparameters in your architecture such as number of layers, number of neurons.

Now, feel free to implement as many of these as possible and present a detailed analysis of your findings as before. Present the same details as before, train and test accuracy over epochs, number of parameters in your model, loss value over epochs, your architecture and details of other tricks you employed.

Results

You need to provide two plots for both training and testing between:

- Accuracy and epoch
- loss and epoch