

# Behavioral Cloning Project

---



## Overview

---

This repository contains starting files for the Behavioral Cloning Project.

In this project, you will use what you've learned about deep neural networks and convolutional neural networks to clone driving behavior. You will train, validate and test a model using Keras. The model will output a steering angle to an autonomous vehicle.

We have provided a simulator where you can steer a car around a track for data collection. You'll use image data and steering angles to train a neural network and then use this model to drive the car autonomously around the track.

We also want you to create a detailed writeup of the project. Check out the [writeup template](#) for this project and use it as a starting point for creating your own writeup. The writeup can be either a markdown file or a pdf document.

To meet specifications, the project will require submitting five files:

- model.py (script used to create and train the model)
- drive.py (script to drive the car - feel free to modify this file)
- model.h5 (a trained Keras model)
- a report writeup file (either markdown or pdf)
- video.mp4 (a video recording of your vehicle driving autonomously around the track for at least one full lap)

This README file describes how to output the video in the "Details About Files In This Directory" section.

## Creating a Great Writeup

---

A great writeup should include the [rubric points](#) as well as your description of how you addressed each point. You should include a detailed description of the code used (with line-number references and code snippets where necessary), and links to other supporting documents or external references. You should include images in your writeup to demonstrate how your code works with examples.

All that said, please be concise! We're not looking for you to write a book here, just a brief description of how you passed each rubric point, and references to the relevant code :).

You're not required to use markdown for your writeup. If you use another method please just submit a pdf of your writeup.

## The Project

---

The goals / steps of this project are the following:

- Use the simulator to collect data of good driving behavior
- Design, train and validate a model that predicts a steering angle from image data
- Use the model to drive the vehicle autonomously around the first track in the simulator. The vehicle should remain on the road for an entire loop around the track.
- Summarize the results with a written report

## Dependencies

This lab requires:

- [CarND Term1 Starter Kit](#)

The lab environment can be created with CarND Term1 Starter Kit. Click [here](#) for the details.

The following resources can be found in this github repository:

- `drive.py`
- `video.py`
- `writeup_template.md`

The simulator can be downloaded from the classroom. In the classroom, we have also provided sample data that you can optionally use to help train your model.

## Details About Files In This Directory

### `drive.py`

Usage of `drive.py` requires you have saved the trained model as an h5 file, i.e. `model.h5`. See the [Keras documentation](#) for how to create this file using the following command:

```
model.save(filepath)
```

Once the model has been saved, it can be used with `drive.py` using this command:

```
python drive.py model.h5
```

The above command will load the trained model and use the model to make predictions on individual images in real-time and send the predicted angle back to the server via a websocket connection.

Note: There is known local system's setting issue with replacing "," with "." when using `drive.py`. When this happens it can make predicted steering values clipped to max/min values. If this occurs, a known fix for this is to add "export LANG=en\_US.utf8" to the `bashrc` file.

### Saving a video of the autonomous agent

```
python drive.py model.h5 run1
```

The fourth argument, `run1`, is the directory in which to save the images seen by the agent. If the directory already exists, it'll be overwritten.

```
ls run1

[2017-01-09 16:10:23 EST] 12KiB 2017_01_09_21_10_23_424.jpg
[2017-01-09 16:10:23 EST] 12KiB 2017_01_09_21_10_23_451.jpg
[2017-01-09 16:10:23 EST] 12KiB 2017_01_09_21_10_23_477.jpg
[2017-01-09 16:10:23 EST] 12KiB 2017_01_09_21_10_23_528.jpg
[2017-01-09 16:10:23 EST] 12KiB 2017_01_09_21_10_23_573.jpg
[2017-01-09 16:10:23 EST] 12KiB 2017_01_09_21_10_23_618.jpg
[2017-01-09 16:10:23 EST] 12KiB 2017_01_09_21_10_23_697.jpg
[2017-01-09 16:10:23 EST] 12KiB 2017_01_09_21_10_23_723.jpg
[2017-01-09 16:10:23 EST] 12KiB 2017_01_09_21_10_23_749.jpg
[2017-01-09 16:10:23 EST] 12KiB 2017_01_09_21_10_23_817.jpg
...
```

The image file name is a timestamp of when the image was seen. This information is used by `video.py` to create a chronological video of the agent driving.

### `video.py`

```
python video.py run1
```

Creates a video based on images found in the `run1` directory. The name of the video will be the name of the directory followed by `'.mp4'`, so, in this case the video will be `run1.mp4`.

Optionally, one can specify the FPS (frames per second) of the video:

```
python video.py run1 --fps 48
```

Will run the video at 48 FPS. The default FPS is 60.

### Why create a video

1. It's been noted the simulator might perform differently based on the hardware. So if your model drives successfully on your machine it might not on another machine (your reviewer). Saving a video is a solid backup in case this happens.
2. You could slightly alter the code in `drive.py` and/or `video.py` to create a video of what your model sees after the image is processed (may be helpful for debugging).

## How to write a README

---

A well written README file can enhance your project and portfolio. Develop your abilities to create professional README files by completing [this free course](#).