Master Thesis summary

732A64-Master Thesis

Andreas Christopoulos Charitos
andch552

October 26, 2020

Contents

During this paper we will present the results obtained so far alongside with critic and future steps.

Lets consider again the questions of interest during this project

Role of derivatives

What is the relation of the derivatives in the classification? We are interested to know if a certain derivative or derivative combination gives better classification

Data augmentation

What kind of 3D augmentation (e.g. rotations, scaling of each volume) are most important in the 3D CNN training?

Architecture

What is the best architecture of the 3D CNN? We wish to know how many layers to use and number of filters in each layer in the model architecture.

1 Training and Evaluation

To begin with, at first we decide to create some combinations of derivatives and test which combination or single derivative can result a better classification. The choice of the combinations of interest include i) one combination with only the dual regression derivative (10-channels) ii) one combination with all the derivatives except dual regression (9-channels) and iii) one combination with all the derivatives (19-channels). In order to increase the size of the datasets a series of augmentations is performed with the following methods considered

Augmentations

1. rotation
2. flip
3. center crop
4. blur
5. elastic deformation

Following up, the discovery of the best CNN architecture is performed within the concept of grid search with predefined discrete values for the parameters of interest. During this project the following hyper-parameters are considered

Hyper-parameters

- number of convolutional layers $\in [3, 4, 5]$
- number of filters int the convolutional layers $\in [32, 64, 128]$
- dropoout rate $\in [0.4, 0.5, 0.6]$

- dense nodes $\in [128, 256, 512, 1024]$

The process of evaluating each combination is performed with MC dropout were the model is exposed to a hold-out dataset for 100 times and the average accuracy/loss is reported as test accuracy/loss

2 Results so far

This section is dedicated to the presentation of the results obtained so far with some illustrations. The proper visualization in order to present the combinations tested for each architecture is suggested to be a parallel coordinate plot. For the results presented in the following sections the hyper-parameters1 previously presented were used with the exception in the number of convolutional layers where only 4 layers tested.

2.1 Dual Regression Derivative

One of the first combination of derivatives that tested was dual regression derivative. The figure 1 shows the obtained pair accuracy/loss for each combination

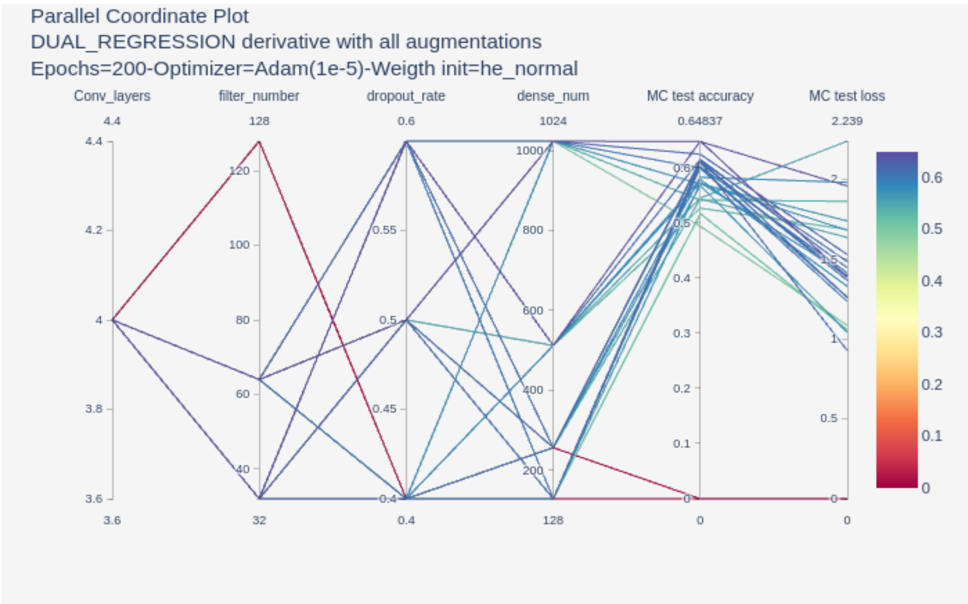

Figure 1: Parallel coordinates plot dual regression

As we can see from the above plot the best accuracy was roughly 0.65% and achieved with the following settings

ConvLayers	nFilts	dropoutRate	denseNodes	loss	accuracy
4	32	0.6	512	0.13875	0.6483

The next plot provides an illustration of the MC dropout evaluation

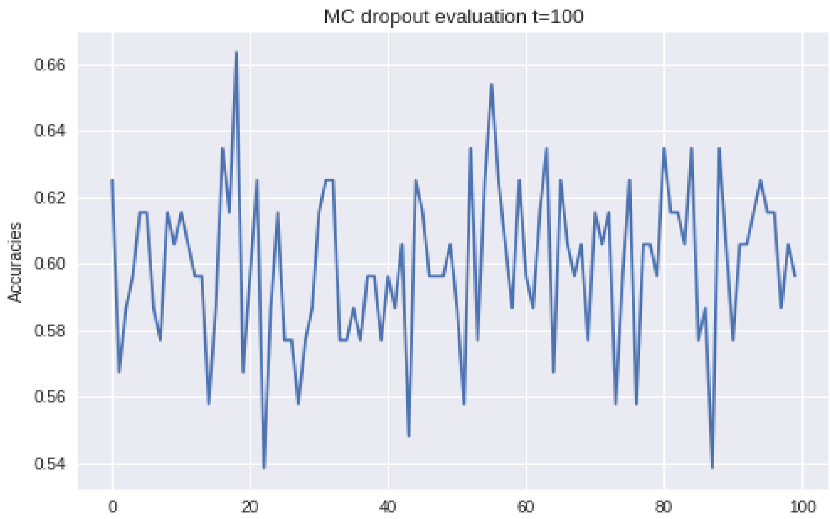

Figure 2: MC evaluation for dual regression

2.2 All 9 Derivatives except Dual Regression

The next combination tested was all 9 derivatives combined without dual regression. The figure 3 shows the obtained pair accuracy/loss for each combination

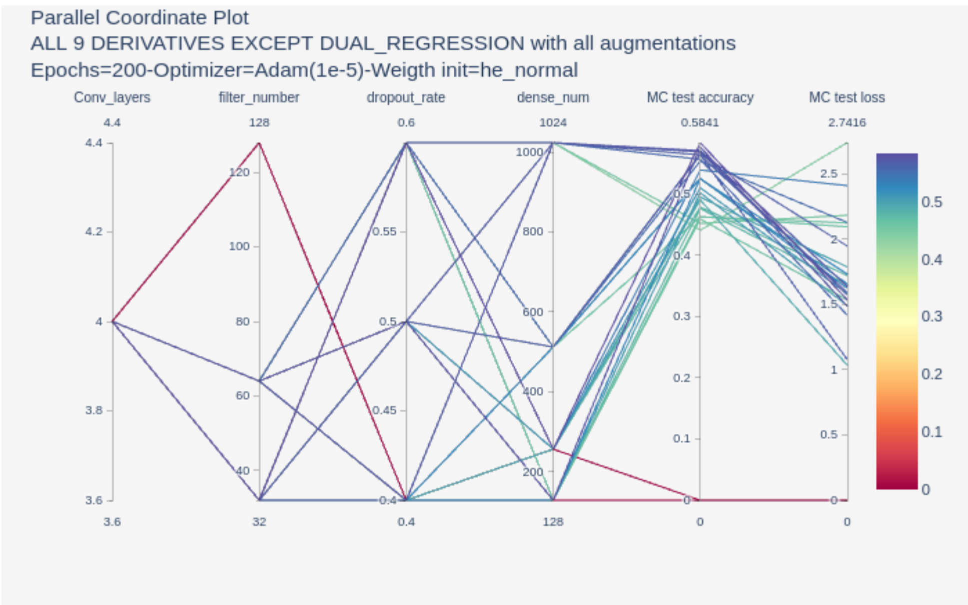

Figure 3: Parallel coordinates plot 9 derivatives

As we can see from the above plot the best accuracy was roughly 0.58% and achieved with the following settings

ConvLayers	nFilts	dropoutRate	denseNodes	loss	accuracy
4	32	0.6	256	0.15357	0.5841

The next plot provides an illustration of the MC dropout evaluation

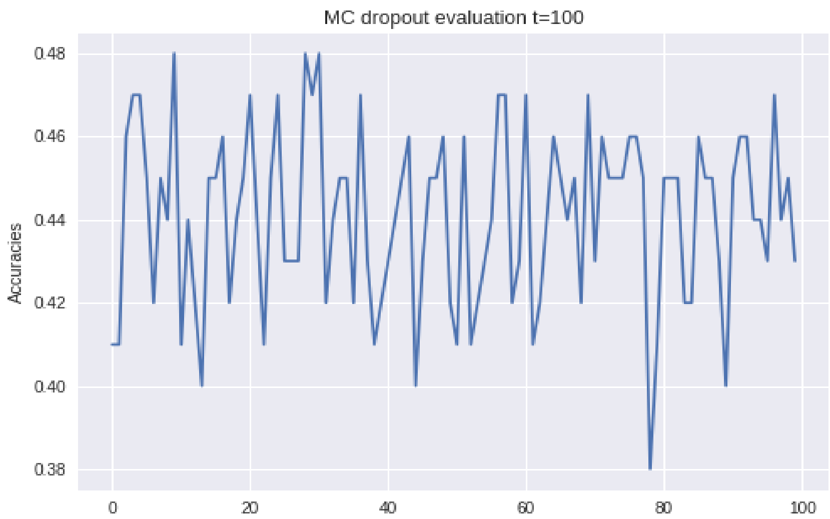

Figure 4: MC dropout 9 derivatives

3 Critic and Future plan

In this section we provide a critic on the methods applied and an overview of the future work to be implemented. The results presented in the previous section have been applied with the settings mentioned in the introduction besides the ConvLayers hyper-parameter where only 4 layers where tested. This was a decision made in order to test the implementation and the computational time. The results seem to be consistent and the future work involves the following steps

Next steps

- test the combinations of interest with all and some augmentations
- perform the training using the entire search space
- investigate the performance of another activation function (PRELU)