

# Homework 06 – Training

Arthur J. Redfern  
[arthur.redfern@utdallas.edu](mailto:arthur.redfern@utdallas.edu)

---

## 0 Outline

- 1 Reading
- 2 Theory
- 3 Practice

## 1 Reading

### 1. Training

Motivation: understand xNN training

[https://github.com/arthurredfern/UT-Dallas-CS-6301-CNNs/blob/master/Lectures/xNNs\\_060\\_Training.pdf](https://github.com/arthurredfern/UT-Dallas-CS-6301-CNNs/blob/master/Lectures/xNNs_060_Training.pdf)

Complete

The following papers are part of a very recent wave of results that challenges relatively recently (or still) held beliefs. Some of these papers are quite dense and it's easy to get lost in the math (don't get dismayed, this happens to everyone). For our purposes here, focus on the text and key results.

- 2. **[Optional]** Dynamical isometry and a mean field theory of CNNs: how to train 10,000-layer vanilla convolutional neural networks

<https://arxiv.org/abs/1806.05393>

Complete

- 3. **[Optional]** Rethinking ImageNet pre-training

<https://arxiv.org/abs/1811.08883>

Complete

- 4. **[Optional]** How does batch normalization help optimization?

<https://arxiv.org/abs/1805.11604>

Complete

5. [Optional] Fixup initialization: residual learning without normalization  
<https://arxiv.org/abs/1901.09321>

Complete

6. [Optional] Image classification at supercomputer scale  
<https://arxiv.org/abs/1811.06992>

Complete

## 2 Theory

None

## 3 Practice

3. Training large networks from scratch with limited resources and / or limited data sets is not fun. Recognizing this, transfer learning is quite popular for vision applications with pre trained encoders and language applications with pre trained language models. To gain more experience with transfer learning, work through the following example:

- [https://pytorch.org/tutorials/beginner/transfer\\_learning\\_tutorial.html](https://pytorch.org/tutorials/beginner/transfer_learning_tutorial.html)

Note that you will have to write some code as indicated to acquire the dataset and execute the script in Google Colab.

Complete