SCUOLA
NORMALE
SUPERIORE

UNIVERSITÀ
DI PISA

Deep Learning Lab

Introduction to the Lab activities

Master Big Data 2021

Andrea Cossu
→ andrea.cossu@sns.it
→ https://andreacossu.github.io/

Ph.D. student in Data Science
@ Scuola Normale Superiore
@ University of Pisa
working on *Continual Learning*

Member of PAI lab and CIML group
ContinualAI Team Lead

Calendar

- Monday, April 19th, 14-18
- Monday, April 26th, 14-18
- Tuesday, May 4th, 14-18
- Monday, May 10th, 14-18
- Monday, May 17th, 14-18
- Microsoft Teams, DL channel (+ separate rooms when needed)

DL Lab tools

- Keras
  - + Tensorflow
- Colab
- Matplotlib, numpy, scikit-learn…

DL Lab outline

- Practical Deep Learning with Keras (and Tensorflow)
- Deep Learning areas
  - Tabular data
  - Computer Vision
  - Sequential data processing (audio, natural language…)
  - Generative models
  - Graph structured data
- You will see more stuff than you actually need for the project!

DL Lab objectives

- Understand the main *principles* behind a Deep Learning library
  - learn one, master all
- Autonomously navigate a deep learning library
  - first google it, then ask it
- Use a deep learning library to solve a specific problem/application
  - aka, fun (and/or $$$)
- We will go **slowly**!
  Most of the things you need are already implemented :)

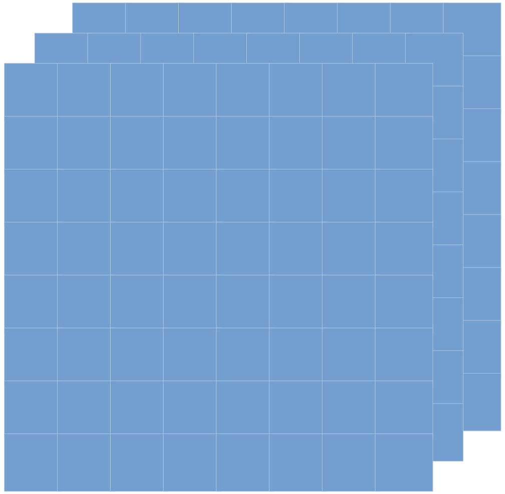
Final Project

- Verify that you acquired the objectives (theory + lab)
- Address a new problem from a known area **your own way**
  - you can use whatever code you like
  - you have to understand it, first
  - so that you can justify your design choice (why this? Have you tried that?)

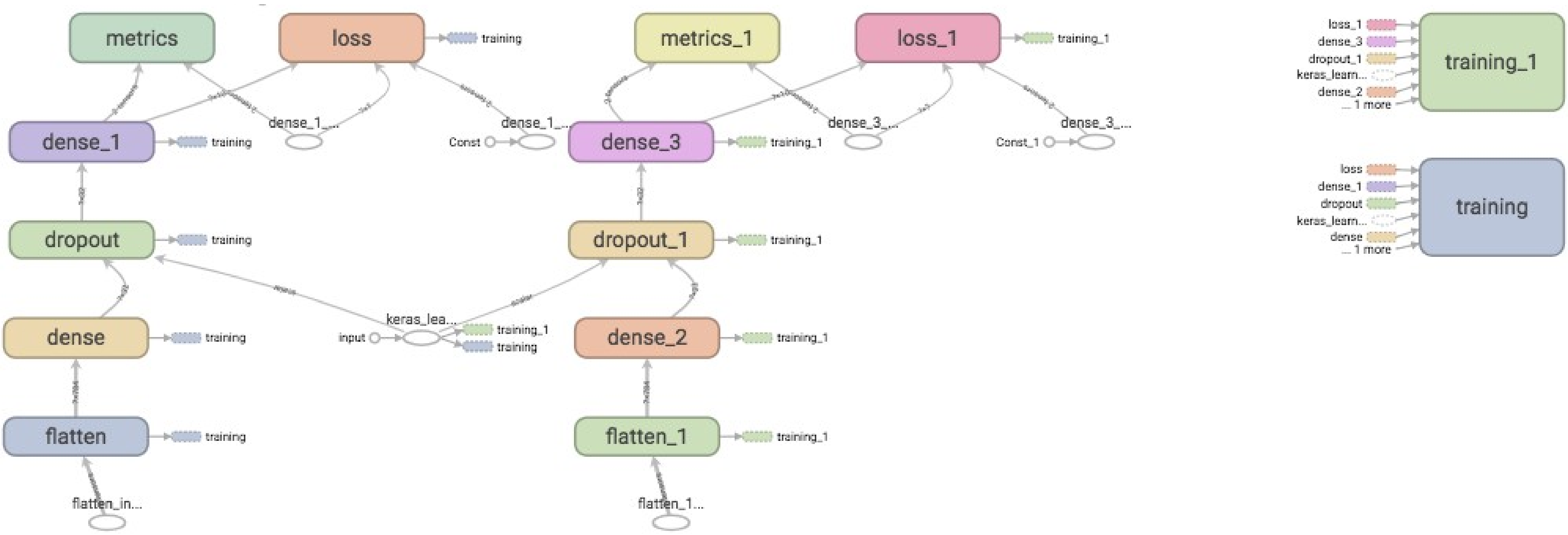
Practical DL
(before digging deeper into the code)

Tensors (the computer science way)

- Multi-dimensional array filled with homogenous values
- They have (at least)
  - A shape/size of n dimensions
    (D1 x D2 x … x Dn)
  - A (data)type: float32, int32, bool…
- Manipulate tensors with common math operators



Computational Graph



https://www.tensorflow.org/tensorboard/graphs

Tensorflow

- Low level API to build DL models, ML pipelines…
- Deal directly with tensors, build models by nesting / composing different modules (e.g. layers)
- Manipulate gradients, control single variables
- Powerful, but more complex

Keras

- High-level API to build DL models, ML pipelines…
- Deal directly with entire modules (e.g. layers)
- Fast prototyping of common paradigms (training loops, dataset manipulation…)
- Powerful and easier…
  - Unless you have to build specific DL models (e.g. change training loop requires mixing Keras + Tensorflow)
- Sometimes, high-level means harder to debug. Each function performs lots of computation, which is hidden to the user.

PyTorch

- Tensorflow equivalent
  - better organized documentation
  - Less oriented to production – but it is changing fast
- High level API are available (e.g. FastAI, Lightning…)
- You don’t want to mix the two in the beginning

Monitoring tools

- You should always have a look at *how* your training is progressing
- Needed to understand which DL tricks to apply to improve performances
- Tensorboard (shipped with Keras), Weights and Biases (free for open-source projects), many others.

Let’s code, shall we?