

A deep-learning neural network for image recognition

A working model

Filippo Biscarini
Senior Scientist
CNR, Milan (Italy)

Nelson Nazzicari
Senior Scientist
CREA, Lodi (Italy)



Objectives

- we start by **showing a working deep learning model for image recognition** (a task at which DL is very good!)
- the objective is to give you some ideas of **what DL is about**
 - no worries if you don't understand everything
 - we'll delve in details in later sessions
- you'll get some **basic intuition of what DL is and how it is structured**



A first working example

- **MNIST** (Modified National Institute of Standards and Technology) database → large collection of **handwritten digits** [more info [here](#)]
- Commonly used to train machine learning models for image recognition
- The aim is to use this database to build a **first deep learning model** for **image recognition**



938209230051315126388691200206341421129960420584193908892612097809
313373106976415247632445737111305533143053706569002152179128231919103
51145930213259432749828549408581849853902820606104811250234342360214
131259348447656101014370197853737516963081900369215677373705453027398
9365366630036363164352965581400861068110401260132687240306356439241
69323797263650014431782216140321614116575140921430925579147675133559
16570280239076694471481820259719135009987525160154101064033878640461
7711157713318992451400380933898206657781236236267214878587897415779
54175611062948264527806612933339900964013343219783280882263928025846
4112710440824467152057718569627371683222684255894332959993019432984
52365156249670241245785168913358702930791093486205447004205020948063
03148274990624736919137421754134673872833106648116548161746711055120
50216315273859182206132616624780824045469217931727638260687232164218
9835674458932709880039098955783937372477328002231793333510693276361
570717593102799694741144880240300328006622439210572870477546594386484
232808297679004206643390473220007459890131540557083946565901585596584
19068771986571010834771309603818920308082651547894010698966356607646
2610269719587006164486233139498521909418862145005122797818737719711
993134195543933585065182689228548926345297173388211237847227878821826
13584885257161838001030240866217943373391184352720356988960609038933
95526953473046294062710391260437194095988305979834494042191670142844
9075742399902521383316760720059828713448821675884571111932567125167
42479848030788394738160872116265818208277827879376862207220652365586
07878692388651132606059910221908809321018038780686908361338473573997
8496134383667562014171867802714610405571749234093018462455394684700
4569887658228301671514155563834986151901055024627410750635089579701
51934322315133581179476371807918112600110961587734707614111846809220
27659122656690746116557859529071323716302942781602824722905377914333
9588203603195374121829552055891377400423768895363898783507800802044
28348641950385962735759452352928985143948094824111977942974139130361
1062402914001952524166895109706300062061127985295280813353392947240872
286422146548996820052412894944731073136654461768716917866663016351589
18807591254682761098394058957291108903040117310864844076829554562194
7158062653096899279609554625797748954817633813374940578579748767350
3518091908102777377421967250574472862114127938670714419004611678421
60436578626808776518823031802950231948251580184227300017994289532
830 Credits: <http://petr-marek.com/> 90087315613112792544385270783368528529244997147

A first working example

- From the MNIST dataset
 - **60,000 images for training**
 - **10,000 images for testing**
 - Reference: <http://yann.lecun.com/exdb/mnist/>

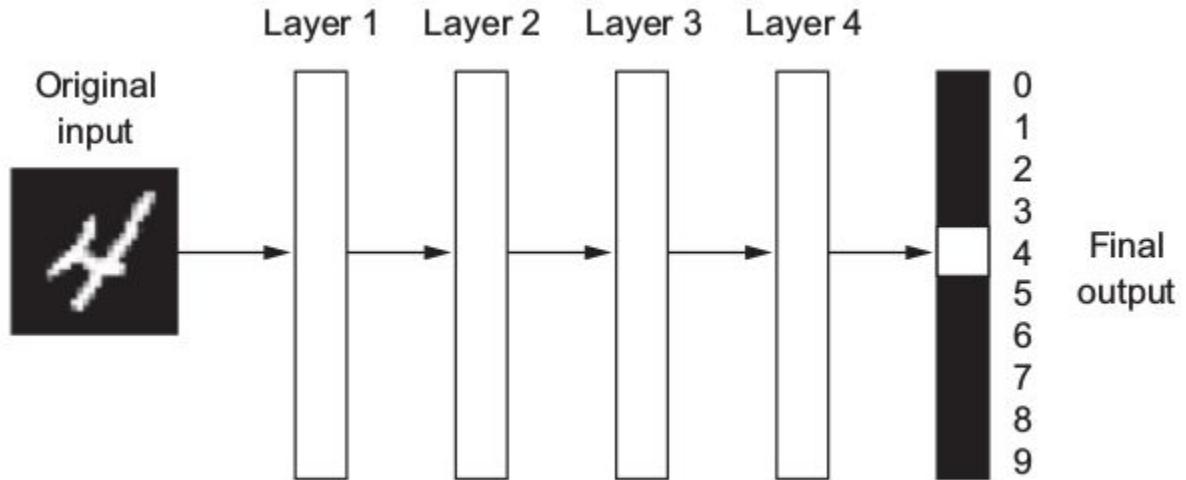


A first working example

- From MNIST
 - 60,000 images for training
 - 10,000 images for testing
 - Reference: <http://yann.lecun.com/exdb/mnist/>
- 1) Step 1: **train the deep learning model**
 - 2) Step 2: get **predictions** (recognize images/handwritten digits) on **test data**
 - 3) Step 3: measure the **accuracy of prediction**



Handwritten digit recognition

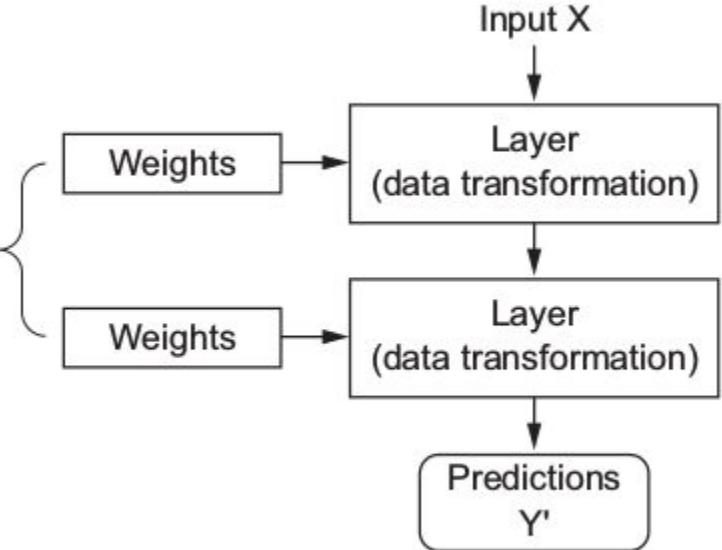


From François Chollet



Model diagram

Goal: finding the right values for these weights



From François Chollet



The needed tools

- Python (3)
- Interactive Python Notebook (.ipynb file) → Jupyter notebooks
- [Google colab](#)
- Keras (wrapper around Tensorflow) [more on this later]



A first working example - components

1. SETUP

- import libraries
- configure parameters

2. DATA MANAGEMENT

- load MNIST data
- data (images) preprocessing

3. MODEL

- build
- compile
- train
- test



A first working example - components

1. SETUP

- import libraries
- configure parameters



The standard part

2. DATA MANAGEMENT

- load MNIST data
- data (images) preprocessing



The boring part

3. MODEL

- build
- compile
- train
- test



The cool part



A first working example - steps (real world)

1. SETUP
 - import libraries
 - configure parameters
 2. DATA MANAGEMENT
 - load MNIST data
 - data (images) preprocessing
 3. MODEL
 - build
 - compile
 - train
 - test
 4. RINSE AND REPEAT
- The standard part
- The boring part
- The cool part
- The professional part



Let's do it!

1. “Black box”
2. Decomposing the model



1- the Black Box

- Training the model
- Getting prediction accuracy on test data
 - go on your computer/server
 - open a terminal
 - run **keras.mnist_train.py**
 - run **keras.mnist_test.py**



2- decomposing the model

- chunk-by-chunk training and testing
- interactive Jupyter notebook
 - **day1_code01_keras_MNIST.ipynb**
 - day2_code00_keras_MNIST_detailed.ipynb [tomorrow]

