



THE UNIVERSITY *of* EDINBURGH
informatics

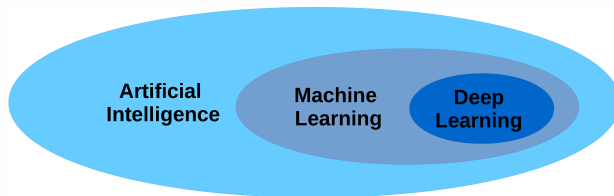
Machine Learning Practical (MLP)

Introduction to MLP and Single Layer Networks

Changjian Li | Pavlos Andreadis

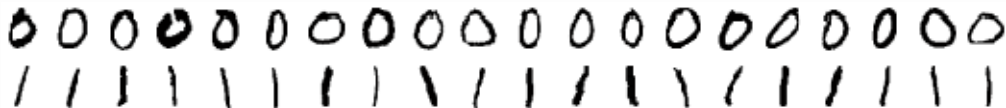
Introduction to MLP

AI, ML and DL



- Artificial Intelligence (AI): building machines to simulate intelligent behavior
- Machine Learning (ML): learning to make decisions by fitting mathematical models to data
- Deep Learning (DL): a type of ML model that learns to represent data in multiple layers of increasing abstraction

AI, ML and Deep Learning?



Task: Classify whether it is a 0 or 1.

AI, ML and Deep Learning?



Task: Classify whether it is a 0 or 1.

Approach 1: Hand-crafted features (e.g. aspect ratio) and rules (e.g. threshold).

AI, ML and Deep Learning?



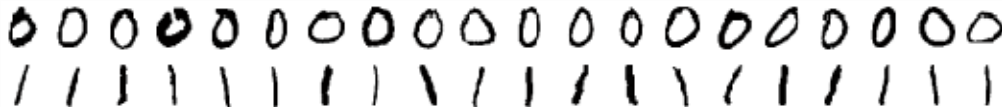
Task: Classify whether it is a 0 or 1.

Approach 1: Hand-crafted features (e.g. aspect ratio) and rules (e.g. threshold).

☺ Easy to understand and debug

☹ It does not scale well to cases with many features (and thresholds)

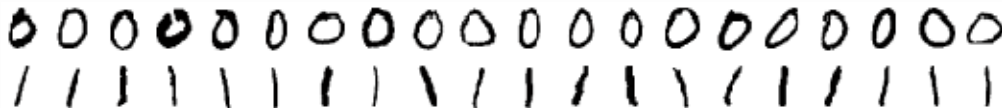
AI, ML and Deep Learning?



Task: Classify whether it is a 0 or 1.

Approach 2: Hand-crafted features and fit a mathematical model (ML).

AI, ML and Deep Learning?



Task: Classify whether it is a 0 or 1.

Approach 2: Hand-crafted features and fit a mathematical model (ML).

- ☺ Scales well to many rules and finds the optimal thresholds
- ☹ Hard to represent high-level features by Hand-crafting them

AI, ML and Deep Learning?



Approach 3: Learn features and rules jointly (DL).

AI, ML and Deep Learning?



Approach 3: Learn features and rules jointly (DL).

☺ Scales well to complex problems

☹ Requires large amounts of data and computation

DL today: Large Language Models

ChatGPT ▾




Create a Renaissance-style painting


Create a workout plan


Explain nostalgia to a kindergartener


Python script for daily email reports

ChatGPT is available for macOS 14+ with Apple Silicon
Get faster access to ChatGPT with the Option + Space shortcut and the floating companion window. [Learn more.](#)

Download ×

 Message ChatGPT 

 User



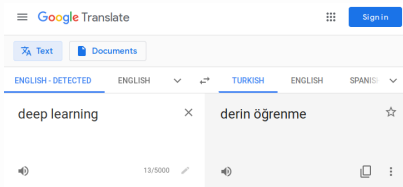
 User

Do you know who drew this painting?

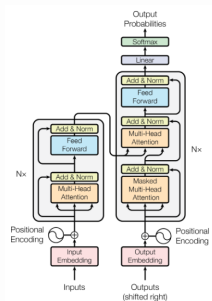
 LLaVA

The painting depicts a dog in a humorous situation, where it appears to be wearing a woman's dress and standing in front of a landscape. Although the artist of the painting is not explicitly identified, it could be a reference to the famous artist Da Vinci, known for his artistic works, including the Mona Lisa. The painting is likely a creative and comical take on the traditional portrait style, replacing a human figure with a dog dressed in a woman's clothing, making it an interesting and amusing visual piece.

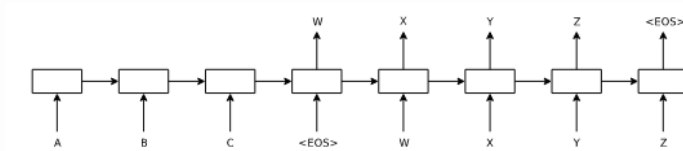
DL today: Machine Translation



Google Translate

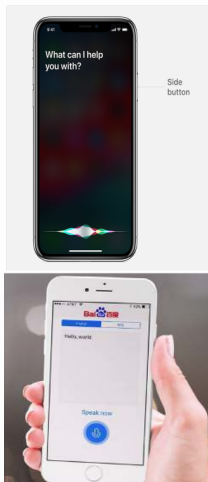


Vaswani et al. 2017

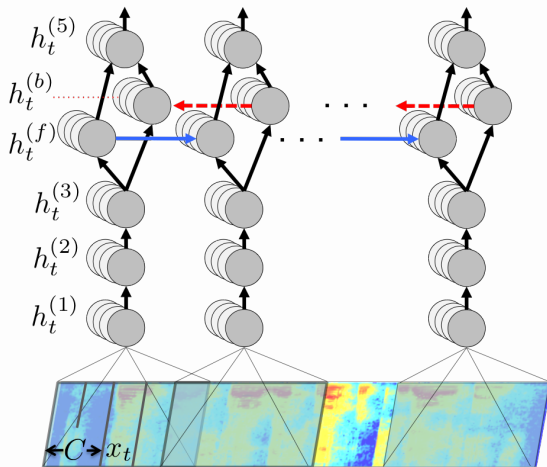


Sutskever et al. 2014

DL today: Speech-to-Text



Siri & Baidu



Hannun et al. 2014

DL today: Vision

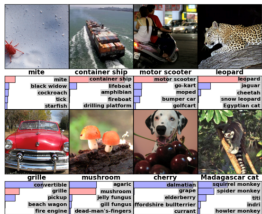


Image classification



Super resolution

SAM uses a variety of input prompts

Prompts specifying what to segment in an image allow for a wide range of segmentation tasks without the need for additional training.



Prompt it with interactive points and boxes.

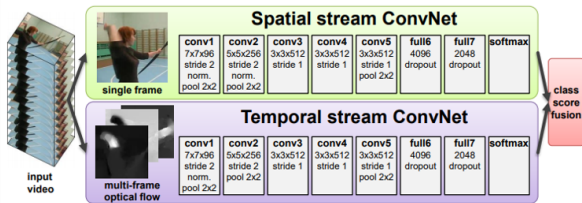


Automatically segment everything in an image.



Generate multiple valid masks for ambiguous prompts.

Semantic segmentation

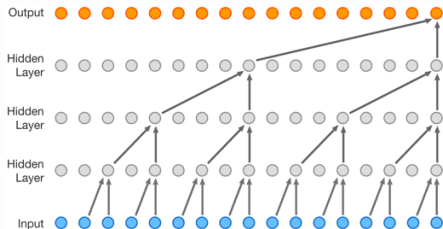


Action classification

DL today: Generative Models



Image generation



Speech synthesis

TEXT DESCRIPTION

An astronaut Teddy bears A bowl of soup

riding a horse lounging in a tropical resort in space playing basketball with cats in space

in a photorealistic style in the style of Andy Warhol as a pencil drawing



DALL-E 2



Image generation from text

Why now?

- Better understanding & algorithms & architectures
- Large (labelled) datasets [WWW, ImageNet, Kinetics]
- More computation power [GPU, TPU]
- Open source tools & models [PyTorch, TensorFlow]
- Open access e-prints [arXiv]

IMAGENET



PYTORCH



MLP – People

- Lecturers: Pavlos Andreadis (Week 1, 3, 7, 11) and Changjian Li (Week 2, 4-6, 8-11)
- TA: Thomas Wong (Course Support), Elchin Asgarov (EdStem)
- Demonstrators (Sem 1), Tutors (Sem 2), Markers (Sem 1 and 2)
- Online Q&A / Forum – Go to Learn course page, then click on "EdStem Discussion"
- Office hours
 - Every week (up till Week 11 inclusive):

Office Hours are drop in (no appointment, and first come first served), both in person (office details below) and online (message the lecturer on Teams).

Office hours should be assumed running, unless we announce an exception for a specific week.
 - Please see Learn > Office Hours, for up-to-date information on Office Hours

MLP – Format

Overall structure:

- Live lectures + recorded lectures (Learn)
- Sem 1: 1 lab/week (starts in week 2)
- Sem 2: 1 tutorial/week
- Assessed by coursework only
- About 6.5 h/week independent working during each semester

How your week might look like (see ‘Course Materials/Week X’ in Learn):

- Read the recommended reading before the lecture
- Attend the live lecture
- Attend the lab/tutorial
- Ask questions on EdStem

MLP – Sem 1 Lectures

Week 1 Introduction and Single layer (shallow) networks

Week 2 Single layer (shallow) networks

Week 3 Multi-layer (deep) networks

Week 4 Generalisation and regularisation

Week 5 Learning algorithms

Week 6 Initialization, Normalization, Pretraining

Week 7 Convolutional networks

Week 8 Sequence modelling: Recurrent networks

Week 9 Sequence modelling: LSTMS

Week 10 Sequence modelling: Transformers

Week 11 ML Practice Design

MLP – Practical Content

- Main focus: investigating deep neural networks using Python
 - Semester 1: the basics
handwritten digit recognition (MNIST)
NumPy, Jupyter Notebook, PyTorch.
 - Semester 2: project-based, focused on a specific task
Projects groups of 2–3 people
Your favorite deep learning library (TensorFlow, PyTorch, etc)
- Approach: implement DNN training and experimental setups within a provided framework, propose research questions/hypotheses, perform experiments, make some conclusions

Coursework

- Four pieces of assessed coursework:
- Semester 1 – using the basic framework from the labs
 1. Interim submission for CW 2
(due Friday 23 October 2026, cohort-level feedback, EdStem/Office-Hours)
 2. Neural Network experiments and report
(due Friday 20 November 2026, worth 50%)
- Semester 2 – group project
 3. Interim report
(due Friday 12 February 2027, feedback only)
 4. Final report
(due Friday 2 April 2027, worth 50%)

Good scholarly practice – remember the University requirement for assessed work.

Requirements

- Programming ability (we will use Python/NumPy)
- Mathematical confidence
- Previous exposure to Machine Learning
- Enthusiasm for Machine Learning

Requirements

- Programming ability (we will use Python/NumPy)
- Mathematical confidence
- Previous exposure to Machine Learning
- Enthusiasm for Machine Learning

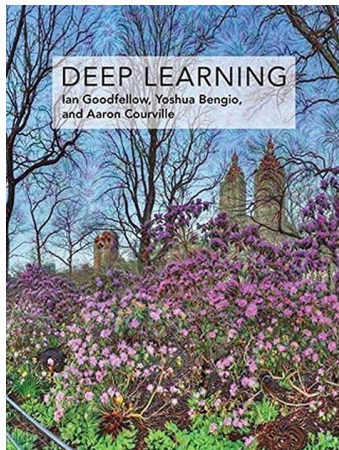


This course is not an introduction to machine learning



Do not take MLP if you do not meet the requirements

Textbook

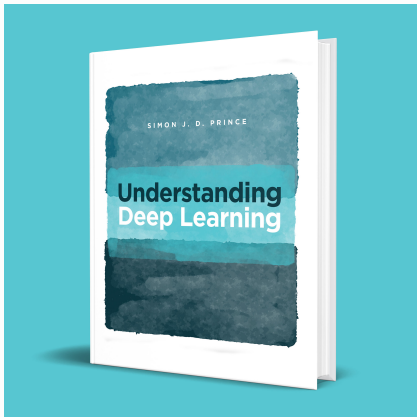


Ian Goodfellow, Yoshua Bengio and Aaron Courville,
Deep Learning, 2016, MIT Press.

- Main textbook
- Comprehensive
- Free online version

<http://www.deeplearningbook.org>

Supplementary Textbook

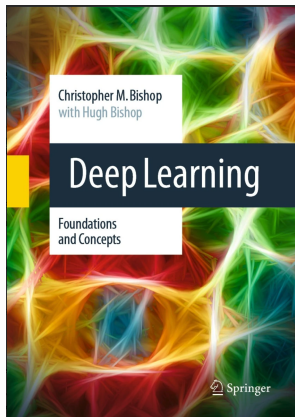


Simon J. D. Prince, *Understanding Deep Learning*, 2023, MIT Press.

- Comprehensive
- Includes more recent advances in DL
- Free online version

<https://udlbook.github.io/udlbook/>

Supplementary Textbook



Christopher M. Bishop, Hugh Bishop, *Deep Learning Foundations and Concepts*, 2024, Springer.

- Comprehensive
- Updated with more recent advances in DL
- Free online version <https://link.springer.com/book/10.1007/978-3-031-45468-4>

MNIST Handwritten Digits



Labs, semester 1

The practical part of MLP is based on a series of labs which explore the material presented in the lectures. The labs are based on the following:

- **Git:** Code and other materials for the labs are available using `git` from a Github repository: <https://github.com/cortu01/mlpractical>.
All necessary git commands will be explained as we go along, but if you have not used `git` before, reading a concise guide is helpful, e.g.
<http://rogerdudler.github.io/git-guide/>
- **Jupyter notebook:** The labs will be presented as Jupyter notebooks, containing both text and code. The first lab includes an introduction to Jupyter notebook.

Labs, semester 1 (cont)

- **Python/NumPy/Matplotlib:** All the code we use and develop in semester uses Python and the NumPy package. This is briefly introduced in first lab, and if you are new to NumPy we encourage you to go through the tutorial linked from the lab
- **mlp:** A NumPy based neural network package designed specifically for the course that you will (partly) implement and extend during the labs and coursework

As explained in the README file on the repository, you need to setup your environment before starting the first lab.

Lab 1: 01_Introduction

After setting up your environment, do the first lab.

The first lab notebook (`01_Introduction.ipynb`) covers:

1. Getting started with **Jupyter Notebook**
2. Introduction to **NumPy** and **Matplotlib** – if you are not familiar with NumPy, then download and follow the Jupyter Notebook tutorial linked from this lab
3. **Data Providers**
 - Modules to load and iterate over data used for training, validating, and testing neural networks
 - `MNISTDataProvider` – class to load and iterate over the MNIST database of handwritten digit images

(Try to do this by the end of week 2)

Any questions?