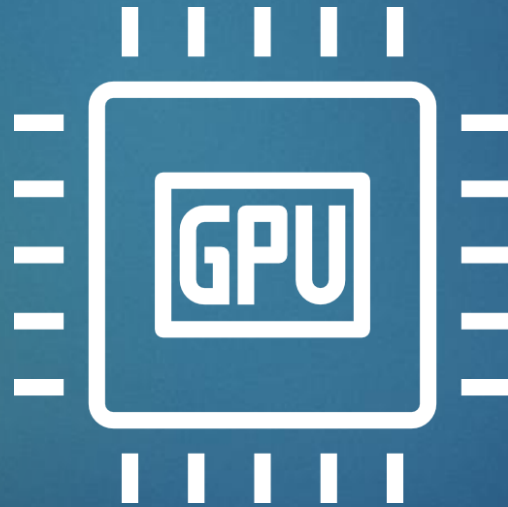# Deep Learning from Scratch

Session #4: Hardware and Platforms

by: Ali Tourani – Summer 2021

Agenda

- ▶ Warm-up and Review

- ▶ Importance of Hardware

- ▶ Google Colab

Warm-up and Review

▶ **Artificial Neural Networks (ANNs)**

    ▶ Bias, perceptron, Activation Functions

▶ **Training the Network**

    ▶ GDA, loss functions and optimization

▶ **Data**

    ▶ Data types, datasets, training and test sets

▶ **Feeding DNNs**

    ▶ Batch, epoch, iteration

    ▶ Hyperparameters, overfitting/underfitting
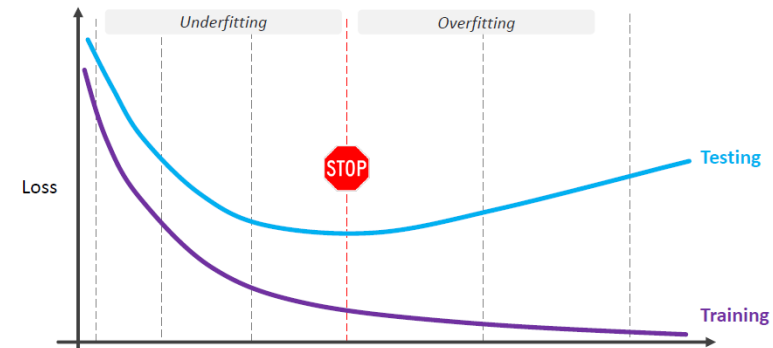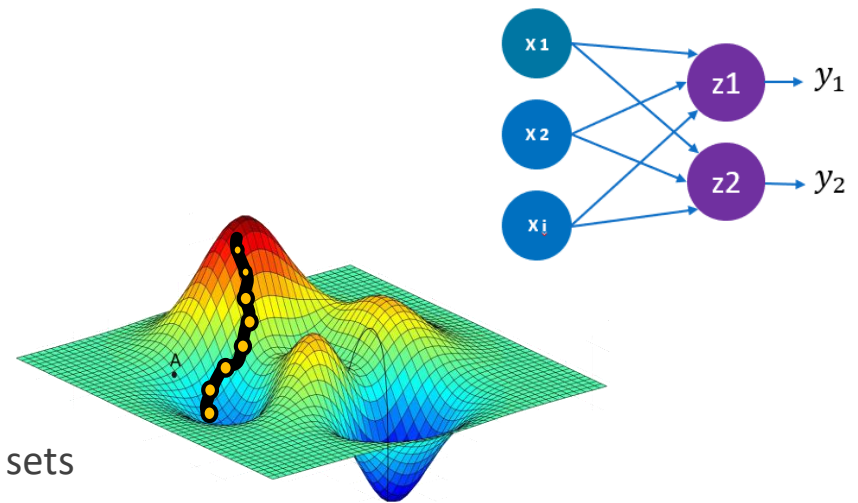
Importance of Hardware

▶ **Main requirements of Deep Learning**

    Big Data

    Powerful Hardware

We are here!   **Efficient Software**

Why is hardware so important?

✔ Deep learning models require datasets with hundreds/thousands of instances

✔ Processing the huge amount of data is <u>impossible</u> with weak resources

✔ Parallelization is a fundamental demand in deep learning

Image source: https://www.hp.com

Importance of Hardware

General Guides

▶ Laptops? Maybe not!

  ▶ Even the most powerful gaming ones

▶ Desktop computers?

  ▶ Equipped with **GPU**

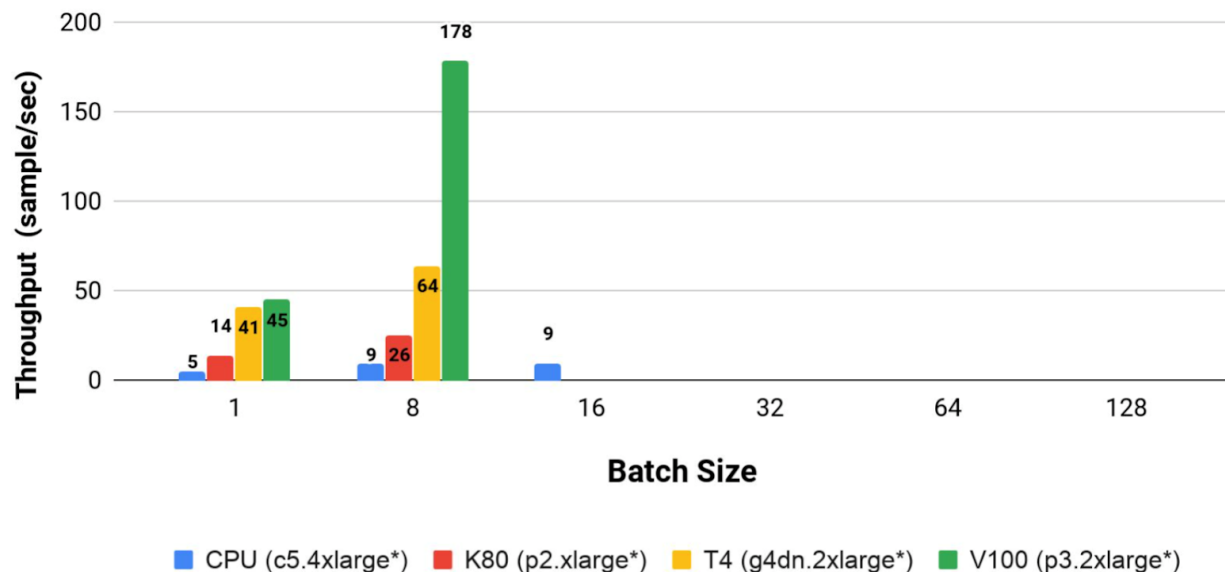**Only CPU**

One GPU

Two GPUs

Four GPUs

Eight GPUs (rack-mount)

Image source: https://www.gamespot.com and https://Sweden.desertcart.com

Importance of Hardware

CPUs vs. GPUs

▶ *Use-case:* throughput evaluation for object detection by DECI

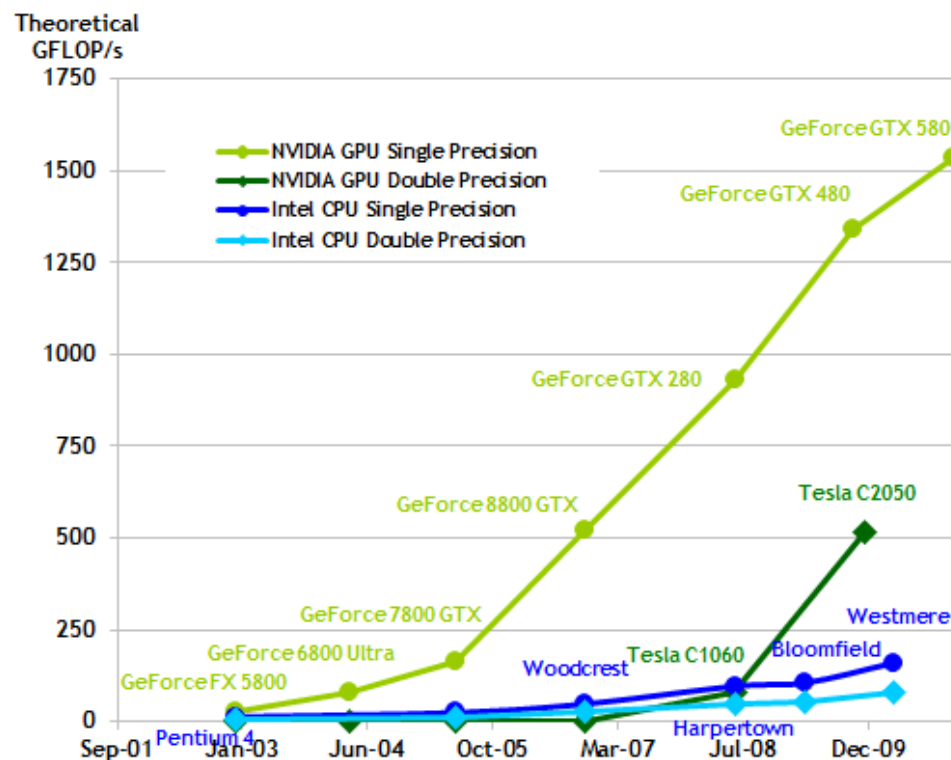*Image source: https://www.pngkit.com*

Importance of Hardware

Why GPUs?

▶ Parallelism with many-core processors

▶ Efficient for Single Program Multiple Data (SPMD)

(tip) You can check my [presentation](#), "an Introduction to CUDA"

Theoretical GFLOP/s chart comparing NVIDIA GPU Single Precision, NVIDIA GPU Double Precision, Intel CPU Single Precision, and Intel CPU Double Precision from Sep-01 to Dec-09, with labels: GeForce FX 5800, GeForce 6800 Ultra, GeForce 7800 GTX, GeForce 8800 GTX, GeForce GTX 280, GeForce GTX 480, GeForce GTX 580, Tesla C1060, Tesla C2050, Pentium 4, Woodcrest, Harpertown, Bloomfield, Westmere.

Image source: https://www.researchgate.net

Google Colab

Why is Google Colab?

▶ A free online cloud-based Jupyter Notebook

▶ Looking for some **free GPUs** for DL practices?

▶ Simply start developing using Python libraries

▶ **Benefits:**

    ▶ A great tool for AI researchers, data scientists, and students

    ▶ Free access to CPUs, GPUs, and TPUs

    ▶ Easy code sharing and zero configuration required

https://colab.research.google.com/

Image source: https://www.colab.research.google.com

Google Colab

Getting Started

▶ See https://colab.research.google.com/notebooks/intro.ipynb

▶ What to expect in a **Colab Notebook**?

This is a Text cell → *A Text Cell*

A Code Cell

```
numberOfParticipants = 25
courseName = 'Deep Learning from Scratch'

print(f'There are {numberOfParticipants} people attended the "{courseName}" course.')
```

Run

```
There are 25 people attended the "Deep Learning from Scratch" course.
```

Output

Google Colab

A great tool for you!

▶ You can have both executable code and rich text documents

▶ You can easily share your work with others

▶ You can load data from Google Drive, Google Sheets, etc.

▶ You can use popular Python libraries like **NumPy**, **matplotlib**, **TensorFlow**, **Panda**, etc.

▶ You can choose GPUs or TPUs as your hardware accelerator
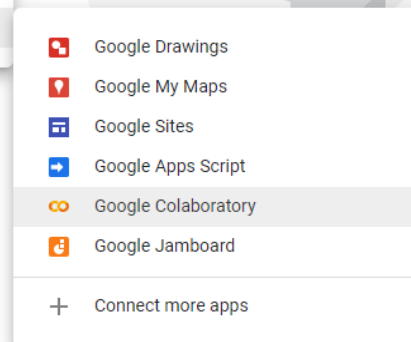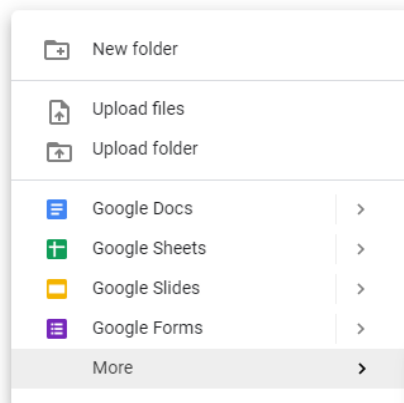
Google Colab

Hardware Accelerators

▶ Switch between TPUs (Tensor Processing Units), GPUs, and CPUs

   ▶ How to get them? Runtime → Change runtime type

Hardware	When to use it?
TPU	- You have Large batches and need the highest possible training throughput
GPU	- You need flexibility and programmability for processing
CPU	- You have large models and need a large memory capacity

Notebook settings

Hardware accelerator
None ⌄ ⑦
None
GPU   tput when saving this notebook
TPU

Cancel   Save
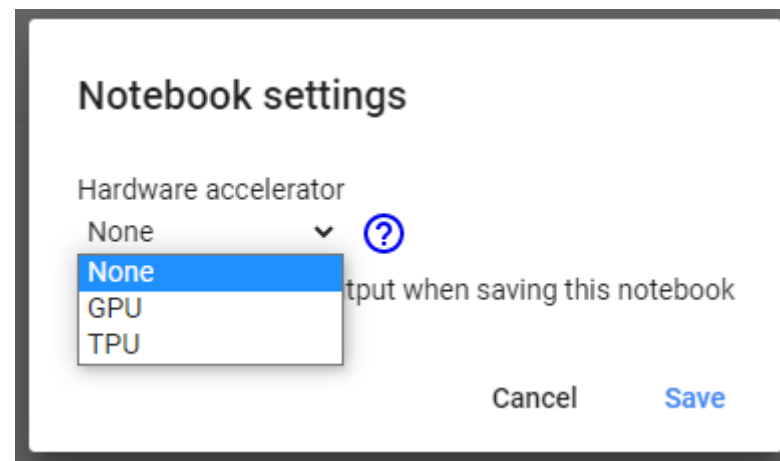
Google Colab

Popular libraries

▶ Simply import and use popular Python libraries

e.g. NumPy + matplotlib for data visualization

```python
import numpy as np
import matplotlib.pyplot as plt

x = np.linspace(0.0, 5.0, 100) # 100 samples between 0 to 5
y = np.sin(4 * np.pi * x) * np.exp(-x)

plt.plot(x, y, 'k')
plt.title('Chart Title')
plt.text(2, 0.65, r'$\sin(2 \pi t) \exp(-t)$') # Chart's inner text
plt.xlabel('X (something)')
plt.ylabel('Y (something)')

plt.subplots_adjust(left = 0.15) # Tune the subplot layout
plt.show()
```
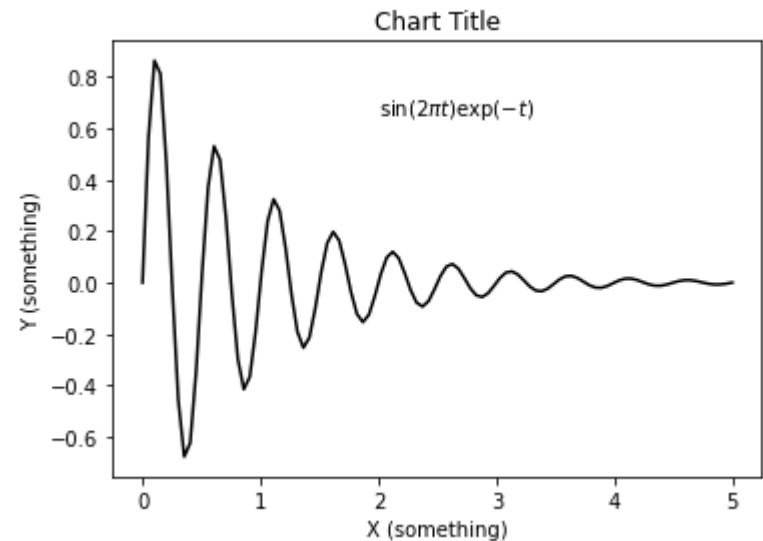
Google Colab

Code Snippets

▶ Some easy-to-use templates to enter common code patterns

  ▶ How to get them? Insert → Code Snippets or just Ctrl + Alt + P

Filter code snippets
Visualization: Bar Plot in Altair →
Visualization: Histogram in Altair →
Visualization: Interactive Brushin... →
Visualization: Interactive Scatter ... →
Visualization: Linked Brushing in ... →

```python
# load an example dataset
from vega_datasets import data
cars = data.cars()

# plot the dataset, referencing dataframe column names
import altair as alt
alt.Chart(cars).mark_point().encode(
  x='Horsepower',
  y='Miles_per_Gallon',
  color='Origin'
).interactive()
```

Google Colab – Code Sample

CO Open in Colab

🚀 **Welcome to Google Colab!**

You can find an intro to Google Colab in this link. Easily create a new Colab Notebook from your Google Drive.

This is a Text cell

```
In [ ]:  # And this is a Code Cell
         numberOfParticipants = 25
         courseName = 'Deep Learning from Scratch'

         print(f'There are {numberOfParticipants} people attending the "{courseName}" course.')
```

```
There are 25 people attending the "Deep Learning from Scratch" course.
```

This is a Code Snippet

Load them from **Insert -> Code Snippets**, or simply **Ctrl+Alt+P**

```
In [ ]:  # load an example dataset
         from vega_datasets import data
         cars = data.cars()

         # plot the dataset, referencing dataframe column names
         import altair as alt
         alt.Chart(cars).mark_point().encode(
            x='Horsepower',
            y='Miles_per_Gallon',
            color='Origin'
         ).interactive()
```

Check full code in GitHub repository

```
Out[ ]:
```

Let's work with popular Python libraries

Agenda

- https://towardsdatascience.com/another-deep-learning-hardware-guide-73a4c35d3e86

- https://colab.research.google.com/github/lexfridman/mit-deep-learning/blob/master/tutorial_deep_learning_basics/deep_learning_basics.ipynb#scrollTo=mH3KKYXSowSe

- https://www.analyticsvidhya.com/blog/2020/03/google-colab-machine-learning-deep-learning/

- https://deci.ai/resources/blog/hardware-for-deep-learning/

- https://analyticsindiamag.com/tpu-vs-gpu-vs-cpu-which-hardware-should-you-choose-for-deep-learning/

Questions?