Fastai v2 (2020) [Jeremy Howard, Sylvain Gugger and Rachel Thomas]

fastbook -> fastai book

Lecture 1

What you don't need, to do deep learning

Myth (don't need)	Truth
Lots of data	Not true always
Lots of math	High school math is sufficient
Lots of expensive computers	We can get what we need for state of art work for free

AGI -> Artificial General Intelligence

Where is deep learning the best known approach

NLP, CV, Medicine, Biology, Image generation, Recommendation systems, Playing games, Robotics

Neural Network: (origin)

1943 Warren McCulloch, a neurophysiologist, and Walter Pitts, a logician, teamed up to develop a mathematical model of an artificial neuron

Mark-I (Frank Rosenblatt)

Marvin Minsky (multilayer pectron) but people ~~misinterp~~ misinterpret

PDP (Parallel Distributed Processing) 1986 MIT

processing units → state of activation → output function → pattern of connectivity → propagation rule → activation rule → learning rule → environment

The age of deep learning (1980s)

universal approximation theorem : stacking up layers with non-linearity in between can allow any model to ~~learn~~ be approximated

Now we have "a machine capable of perceiving, recognizing and identifying its surroundings without any human training or control."

Course strategy

- Play the whole game
- Make the game worth playing
- Work on the hard parts

[top-down approach]

Python

PyTorch

~~FastAI~~ fastai

~~PyTorch~~

fastai used a layered API

$\boxed{\text{Low level}}$ $\boxed{\text{Mid Level}}$ $\boxed{\text{High level}}$

• Removes boilerplate

→ Getting a GPU (first learn and do it in colab)

~~Ren~~

Fine-tuning: A transfer learning technique where the params of a pretrained model are update by training for additional epochs → using a different task to that used for pre training

Right back at the dawn of computing in 1949, an IBM researcher named Arthur Samuel started working on a different way to get computers to complete tasks, which he called machine learning. ~~In his classic way 1962~~

1962 → input, weight → model → results

~~Ite~~ his checkers playing program beated Connecticut state champion

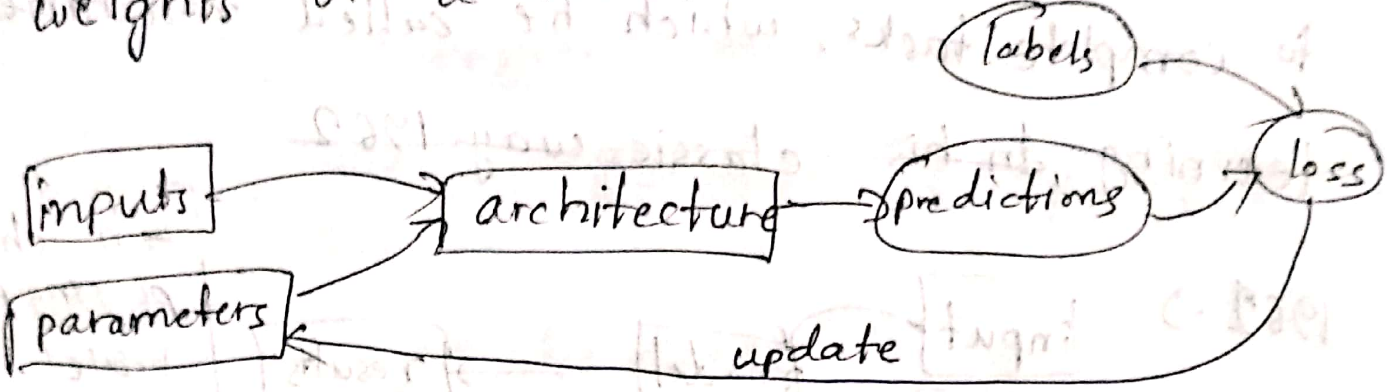
Training a machine learning model

input, weights → model → result → performance $\xrightarrow{\text{update}}$ weights

Later input → model → result

Machine learning: training programs developed by allowing a computer to learn from experience, rather than through manually coding the individual steps.

SGD (Stochastic Gradient Descent) is a completely general way to update the weights of a neural network



~~para~~ model → architecture

~~e~~weights → parameters

predictions → results (independent variables)

measure of performance is called loss

labels (~~var~~ dependent variables)

Limitations

- ~~dat~~ need data
- can only learn from input data (creates bias)
- creates predictions → not recommended actions
- Not enough to just have examples of input data we need labels for that data too

Vision, Text, Tabular, Collaborative filtering

Lecture 2

Classification → aims to predict category or class

Regression → attempt to predict numeric quantities

Metric → a function that measures the quality of a model's prediction using validation set.

error_rate   accuracy

Overfitting is the single most important and challenging issue. Overfitting is like the model is performing well by cheating.

- check with data that you have need seen.

State of DL

Vision (Detection, Classification)

Text (Classification Conversation)

Tabular (High cardinality, GPU) Rapids

Recsys (Predictions, Recommendation)

Multimodal (Labeling, Captioning, Human in the loop)

Other (NLP → Protein)

Catastrophic forgetting → effects after fine-turning

★ High Temperature and High Humidity Reduce Transmission of COVID-19

How might we decide if there's a relationship?

- Pick a "null" hypothesis
- Gather data of independent & dependent variables
- What % of time would we see that relationship by chance?

→ Never do null hypothesis on p-value testing

DataBlock API

get_items :-

Segmentation
Vision, Text, ~~Co~~ Tabular, Collaborative filtering

Lecture 2 [redacted] about fine-tune

Classification → aims to predict category or class
Regression → attempt to predict numeric quantities

Metric → a function that measures the quality of a model's prediction using validation set.

error_rate   accuracy

Importance of validation set:

Overfitting is the single most important and challenging issue. Overfitting is like the model is performing well by cheating.

- check with data that you have need seen.

Transfer learning: Using a pretrained model for a different task to what it was originally trained for

State of DL

Vision (Detection, Classification)
Text (Classification, Conversation)
Tabular (High cardinality, GPU) Rapids
Recsys (Prediction, Recommendation)
Multimodal (Labeling, Captioning, Human in the loop)
Other (NLP → Protein)

fine-tune:
why does transfer learning work? (image)
- deep layer capture and learn structures

Catastrophic forgetting → effects after fine-tuning

★ High Temperature and High Humidity Reduce Transmission of COVID-19 (proven false by Jeremy) (by reverse engineering)

How might we decide if there's a relationship?

- Pick a "null" hypothesis

↓

- Gather data of independent & dependent variables

↓

- What % of time would we see that relationship by chance?

→ Never do null hypothesis or p-value testing

(Concepts ~~of~~ from statistics)

~~DataBlock API~~

~~get_items~~

Turn predictive model into something useful in production

Strategy → Data → Analytics → Implementation → Maintainance

← Identify and manage constraints

- Designing great data products (J Howard et. al.)

DataBlock API

get_items

get_x
get_y (getter)

block[0].create
block[1].create

item_tfms (transforms the items

collate
(DataLoader) [PyTorch DataLoader for better GPU performance]

batch_tfms (batch transforms)