

Intro to Neural Nets

Course Logistics and Introduction



About Me

accenture



MSI at the
ARF



FFG
Forschung wirkt.

Adobe

∞ 3M

 Microsoft

Today's Agenda

1. COURSE LOGISTICS

- Website, schedule, grading and evaluation criteria.
- Course textbook, lecture format, etc.

2. INTERESTING USE CASES

- Frivolous, academic, and practically useful.
- A recent failure, and societal concerns.

3. QUICK INTRODUCTION

- What is a neural network?
- How does it work?





Some Research

arXiv:2412.12610v2 [econ.GN] 4 Feb 2025

Gender Bias and Property Taxes

By Gordon Burtsch[†] & Alejandro Zentner[‡]

Gender bias distorts the economic behavior and outcomes of women and households. We investigate gender biases in property tax hearings. We analyze records of more than 100,000 property tax appeal hearings and more than 2.7 years of associated audio recordings, considering how panelist and appellant genders associate with hearing outcomes. We first observe that female appellants fare systematically worse than male appellants in their hearings. Second, we show that, whereas male appellants' hearing outcomes do not vary meaningfully with the gender composition of the panel they face, those of female appellants do, such that female appellants obtain systematically lesser (greater) reductions to their home values when facing female (male) panelists. Employing a multi-modal large language model (M-LLM), we next construct measures of participant behavior and tone from hearing audio recordings. We observe markedly different behaviors between male and female appellants and, in the case of male appellants, we find that these differences also depend on the gender of the panelists they face (e.g., male appellants appear to behave systematically more aggressively towards female panelists). In contrast, the behavior of female appellants remains relatively constant, regardless of their panel's gender. Finally, we show that female appellants continue to fare worse in front of female panels, even when we condition upon the appellant's in-hearing behavior and tone. Our results are thus consistent with the idea that gender biases are driven, at least in part, by unvoiced perceptions among ARB panelists. Our study documents the presence of gender biases in property appraisal appeal hearings and highlights the potential value of generative AI for analyzing large-scale, unstructured administrative data.

Keywords: Property Tax, Public Finance, Gender Bias, Generative AI, Gender Concordance, Multimodal Large Language Models

[†] Boston University – gburtsch@bu.edu

[‡] UT Dallas – azentner@utdallas.edu

* We would like to thank Emma Wiles, Jetson Leder-Luis, Dokyun Lee, Bin Gu, Patricia Cortés, and Zoe Cullen for helpful comments and suggestions, as well as seminar participants at Southern Methodist University, George Washington University, U Mass-Amherst, and Boston University. Joanna Jia, Heetal Binwani, and Tiffany Zhang provided superb research assistance.

TAKE CAUTION IN USING LLMs AS HUMAN SURROGATES: SCYLLA EX MACHINA*

Yuan Gao
 Questrom School of Business
 Information Systems Department
 Boston University
 Boston, MA 02215
 yuangg@bu.edu

Dokyun Lee
 Questrom School of Business
 Information Systems Department and
 Computing & Data Sciences
 Boston University
 Boston, MA 02215
 dokyun@bu.edu

Gordon Burtch
 Questrom School of Business
 Information Systems Department
 Boston University
 Boston, MA 02215
 gburth@bu.edu

Sina Fazelpour
 Department of Philosophy and
 Khoury College of Computer Sciences
 Northeastern University
 Boston, MA 02115
 s.fazel-pour@northeastern.edu

This Version: Jan 23th, 2025[†]

ABSTRACT

Recent studies suggest large language models (LLMs) can exhibit human-like reasoning, aligning with human behavior in economic experiments, surveys, and political discourse. This has led many to propose that LLMs can be used as surrogates or simulations for humans in social science research. However, LLMs differ fundamentally from humans, relying on probabilistic patterns, absent the embodied experiences or survival objectives that shape human cognition. We assess the reasoning depth of LLMs using the 11-20 money request game. Nearly all advanced approaches fail to replicate human behavior distributions across many models. Causes of failure are diverse and unpredictable, relating to input language, roles, and safeguarding. These results advise caution when using LLMs to study human behavior or as surrogates or simulations.

'She has twelve misshapen feet, and six necks of the most prodigious length;
 and at the end of each neck she has a frightful head with three rows of teeth in each'
 — Homer, *Odyssey* (Describing Scylla)

Introduction

Recent studies report that Large Language Models (LLMs) can exhibit human-like cognitive abilities. These studies demonstrate that LLMs show behaviors that align closely with those of human subjects in seminal experiments from behavioral economics, and responses comparable to those of humans in

[†]Previous Version: Aug 28, Oct 24, and Nov 13th 2024

*We thank seminar participants at the BU, Wharton (Sep 2024), USC, UC Irvine, and Meta. All errors are the author's own.

Some Research

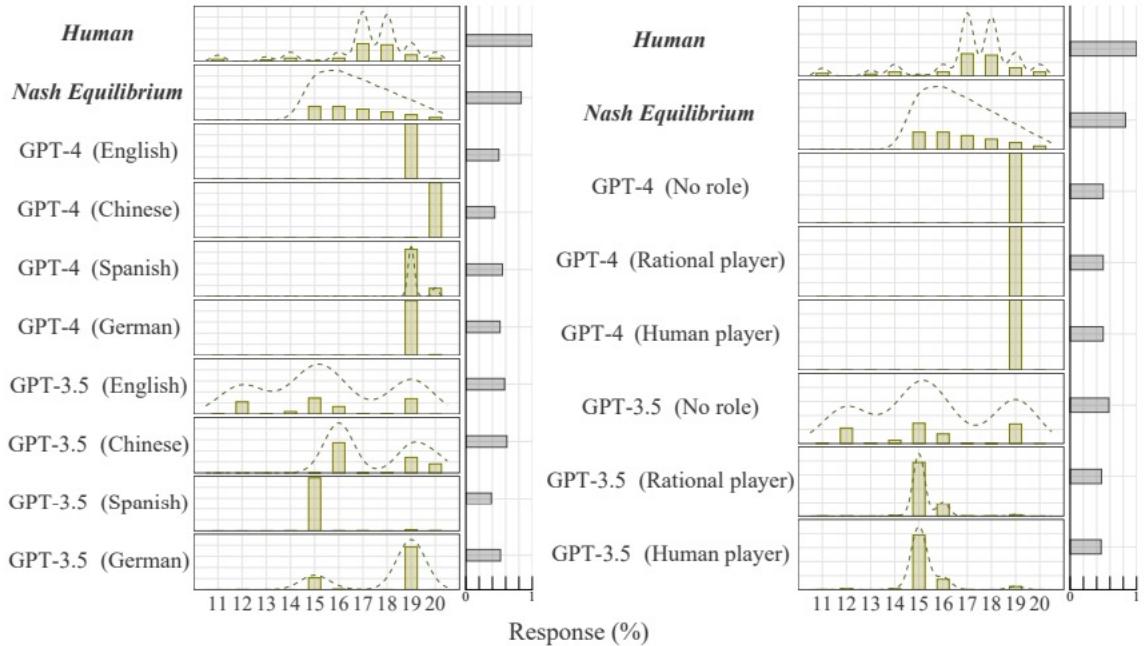


Figure 2: Prompt Brittleness: Roles and Languages. The bar chart on the right shows the similarity between the distribution of different subjects and human subjects, measured by Jensen-Shannon divergence scores. Density plots are omitted for subjects with over 98% of the data concentrated in a single choice to avoid potential misinterpretation.

Grading & Evaluation



Pre-Module

1. Complete all the pre-module reading materials (see **Reading Materials** Section above)
2. Complete the pre-module quiz, which is based on the required pre-module readings (Chapters 1 and 2 of the text).
3. Make sure you are comfortable with the material from the Data Mining in R class (concepts such as Training/Test Sets, Logistic Regression, model error). I will provide no more than a quick review of these pre-requisite materials during class.
4. Create a google account and make sure you can access Google Colab. I recommend that you purchase a Google Colab Pro subscription. You can do this via the settings menu in Google Colab (click the gear icon, then choose Colab Pro).



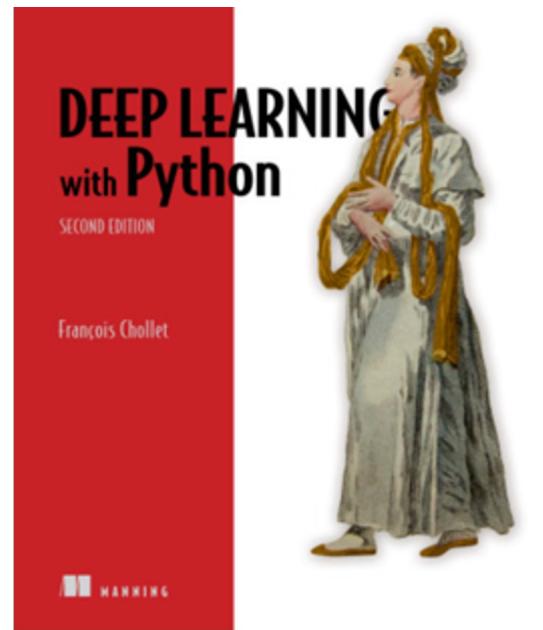
Post Module Exercise

First, read the post-module reading materials (see **Reading Materials** Section). Then, complete the following two post-module assignment items:



- (1) The first part of the assignment is to be done ***independently***. Discussion is allowed, but each student needs to submit his/her own individual write-up. Please submit a single edited .ipynb file (with your name and ID in the file name). (40% of final grade).
- (2) The second part is an open-ended (group) project; you can do it either independently or collaborate with up to two other students (maximum of three students in each group). Please submit a separate single WORD/PDF file with all group members' names and IDs. (35% of final grade)

Course Textbook



Chollet, François. (2021). *Deep Learning with Python (2nd Edition)*.
Manning Publications Co. **ISBN-13: 978-1617296864**.
<https://www.manning.com/books/deep-learning-with-python-second-edition>

Required Software

SOFTWARE CONFIGURATION

- You can access Google Colab at <https://colab.research.google.com>. You should have setup a colab account already.



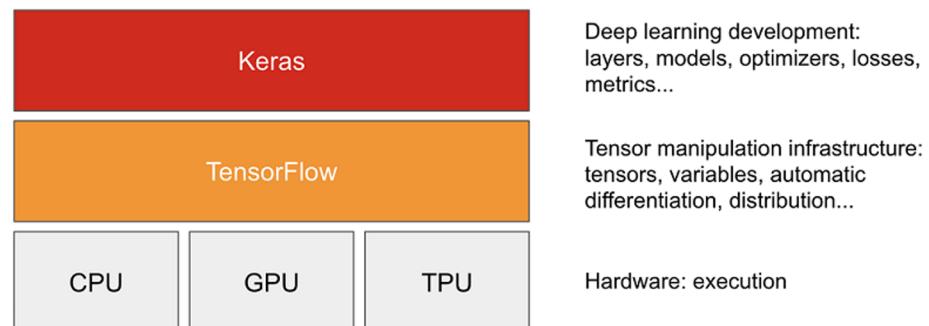
Keras and Tensorflow

1. Tensorflow

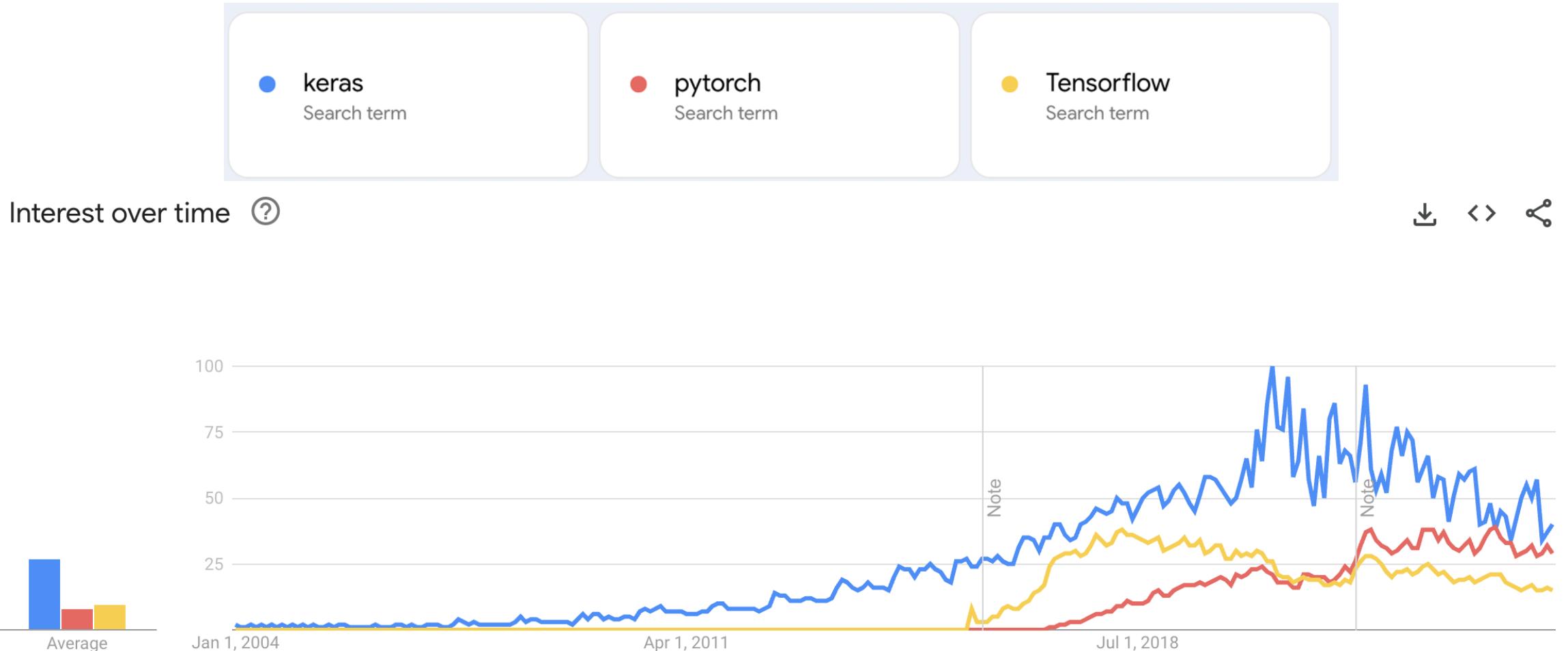
- A Python platform for working with tensors, implementing automatic differentiation, providing access to repositories of (well-known) pre-trained models.

2. Keras

- A higher-level API that wraps common usage patterns with Tensorflow functions, pre-defined loss functions, optimization algorithms, etc.
- Keras simplifies data scientists' interaction with Tensorflow.



Why Keras?



Google Colab

The screenshot shows a Google Colab interface with a Jupyter notebook titled "getting-started-keras.ipynb". The notebook is displayed in a browser window with a tab bar at the top showing the URL: colab.research.google.com/github/GoogleCloudPlatform/cloudml-samples/blob/master/notebooks/tensorflow/getting-started-keras.ipynb?authuser=1#scrollTo=mHF9VCProKJN. The browser also has a toolbar with various icons for navigation and file operations.

The notebook content starts with a section titled "Getting started: Training and prediction with Keras in AI Platform". This section includes a "Table of contents" sidebar on the left with the following items:

- Getting started: Training and prediction with Keras in AI Platform
 - Overview
 - Dataset
 - Objective
 - Costs
- Before you begin
 - Set up your local development environment
 - Set up your GCP project
 - Authenticate your GCP account
 - Create a Cloud Storage bucket
- Part 1. Quickstart for training in AI Platform
 - Get training code and dependencies
 - Train your model locally
 - Train your model using AI Platform
 - Hyperparameter tuning
- Part 2. Quickstart for online predictions in AI Platform

The main content area displays the first section, "Getting started: Training and prediction with Keras in AI Platform". It features a heading, a "Run in Colab" button, a "View on GitHub" button, and three logos: Keras (red square), TensorFlow (orange T), and AI Platform (blue hexagon). Below the heading, there is a detailed description of the tutorial's purpose and the Keras API.

Getting started: Training and prediction with Keras in AI Platform

This tutorial shows how to train a neural network on AI Platform using the Keras sequential API and how to serve predictions from that model. Keras is a high-level API for building and training deep learning models. [tf.keras](#) is TensorFlow's implementation of this API. The first two parts of the tutorial walk through training a model on Cloud AI Platform using prewritten Keras code, deploying the trained model to AI Platform, and serving online predictions from the deployed model. The last part of the tutorial digs into the training code used for this model and ensuring it's compatible with AI Platform. To learn more about building machine learning models in Keras more generally, read [TensorFlow's Keras tutorials](#).

Overview

This tutorial shows how to train a neural network on AI Platform using the Keras sequential API and how to serve predictions from that model. Keras is a high-level API for building and training deep learning models. [tf.keras](#) is TensorFlow's implementation of this API. The first two parts of the tutorial walk through training a model on Cloud AI Platform using prewritten Keras code, deploying the trained model to AI Platform, and serving online predictions from the deployed model. The last part of the tutorial digs into the training code used for this model and ensuring it's compatible with AI Platform. To learn more about building machine learning models in Keras more generally, read [TensorFlow's Keras tutorials](#).

Dataset

This tutorial uses the [United States Census Income Dataset](#) provided by the [UC Irvine Machine Learning Repository](#). This dataset contains information about people from a 1994 Census database, including age, education, marital status, occupation, and whether they make more than \$50,000 a year.

Course

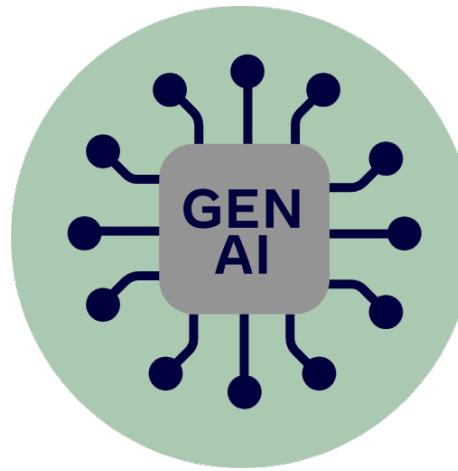


LECTURE AND DISCUSSION

- For each session block over the next two days, I will begin by presenting concepts, describing logic, discussing implementation considerations.

HANDS-ON EXERCISES

- The latter half of each session will be dedicated to hands on examples in Colab notebooks, where I will walk through the code and explain the logic of what we are doing, allow you to ask questions, etc.



Support

USE IT TO HELP YOU LEARN AND PERFORM BASIC TASKS

I expect you to use these tools. However, the way you use them matters. Some valid use cases include ...

- Implementing data munging tasks that you might already understand based on past coursework, e.g., pre-processing text.
- Automatic generation of code comments or documentation.

NOT TO GENERATE SOLUTIONS FROM SCRATCH

If you use these tools as a shortcut to avoid understanding the course material, you will not do well in this class. If we encounter functions and practices that were not taught in the course (e.g., PyTorch code), you will be asked to explain your code to us orally. If you are unable to explain what the code is doing, points *will* be deducted from your grade.

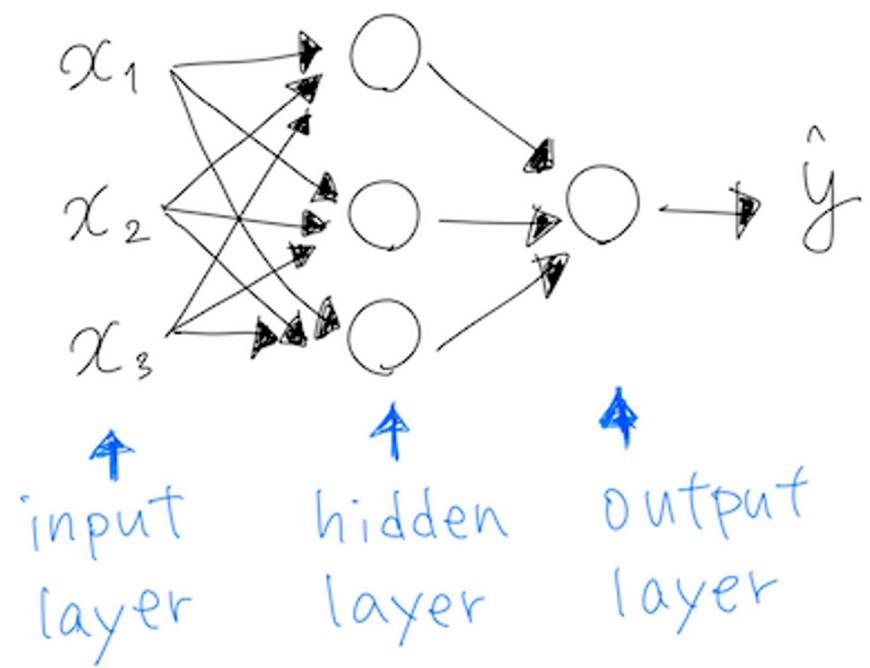
Any Questions?

What is ‘Deep’ (vs. Shallow) Learning?

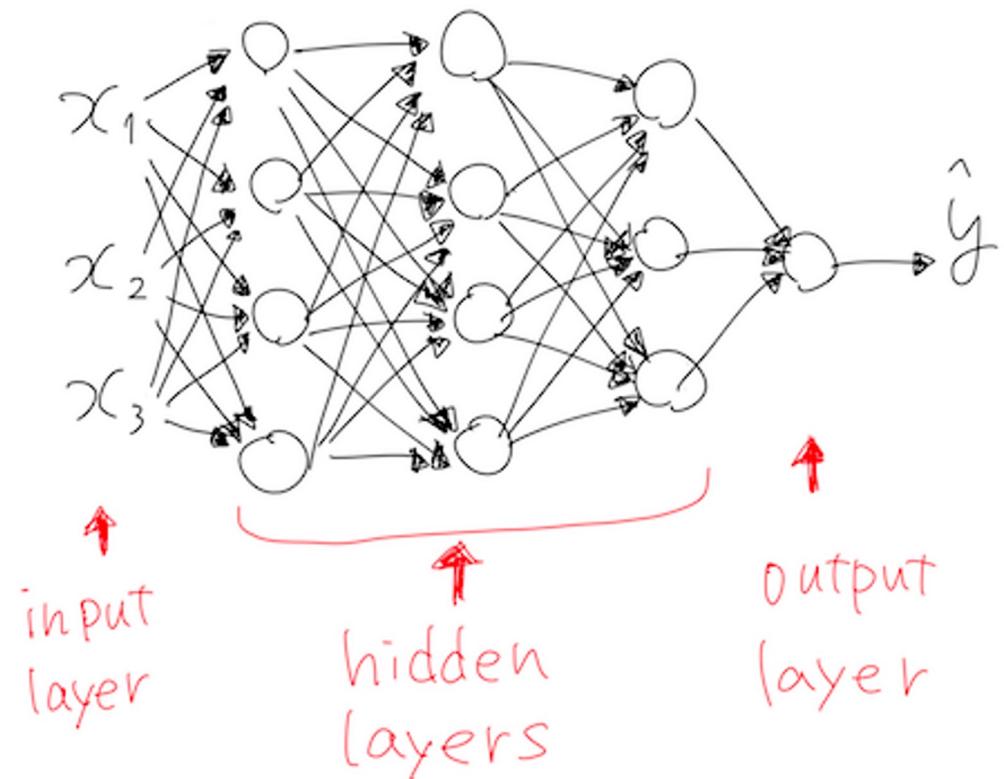
<https://chat.openai.com/chat>

What is 'Deep' Learning?

Shallow Neural Network



Deep Neural Network



Where Deep Learning Started

Communicated by Dana Ballard

Backpropagation Applied to Handwritten Zip Code Recognition

Y. LeCun
B. Boser
J. S. Denker
D. Henderson
R. E. Howard
W. Hubbard
L. D. Jackel

AT&T Bell Laboratories Holmdel, NJ 07733 USA

The ability of learning networks to generalize can be greatly enhanced by providing constraints from the task domain. This paper demonstrates how such constraints can be integrated into a backpropagation network through the architecture of the network. This approach has been successfully applied to the recognition of handwritten zip code digits provided by the U.S. Postal Service. A single network learns the entire recognition operation, going from the normalized image of the character to the final classification.

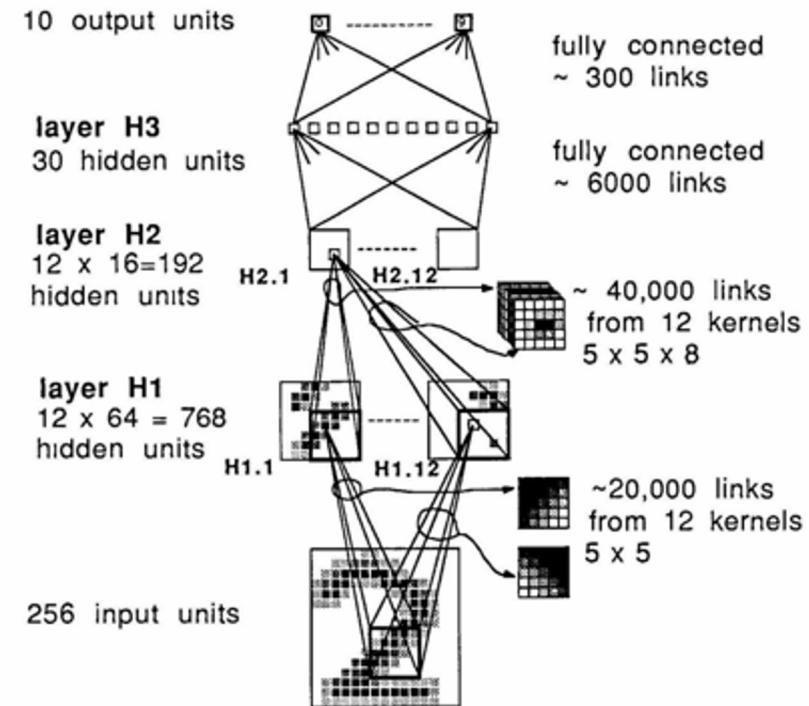


Figure 3 Log mean squared error (MSE) (top) and raw error rate (bottom) versus number of training passes

Then It Shuffled Along for Decades...

What was actually wrong with backpropagation in 1986?

- We all drew the wrong conclusions about why it failed.
The real reasons were:
 1. Our labeled datasets were thousands of times too small.
 2. Our computers were millions of times too slow.
 3. We initialized the weights in a stupid way.
 4. We used the wrong type of non-linearity.

A few years ago, Jeff Dean decided that with enough computation, neural networks might do amazing things.

He built a lot of infrastructure to allow big neural nets to be trained on lots of cores in Google data centers.

THE
ROYAL
SOCIETY





Watch more vid
royalsociety.org 42:50

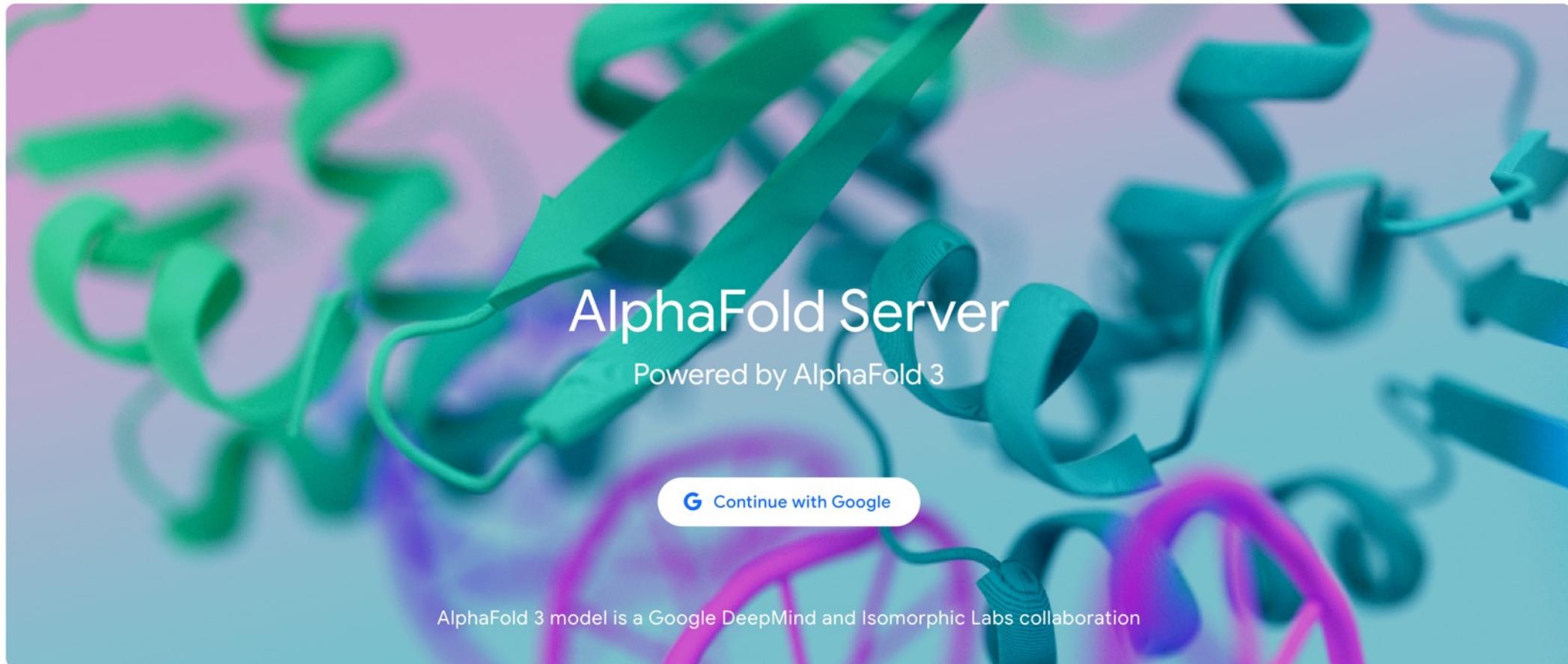
Now...

THE SHIFT

An A.I.-Generated Picture Won an Art Prize. Artists Aren't Happy.

“I won, and I didn’t break any rules,” the artwork’s creator says.

[!\[\]\(eca9dca13a2688a2d5e75f35d4cc16aa_img.jpg\) Share full article](#)[!\[\]\(991b36aa5bd6fb38a409d6026b7522b3_img.jpg\) 1.5K](#)



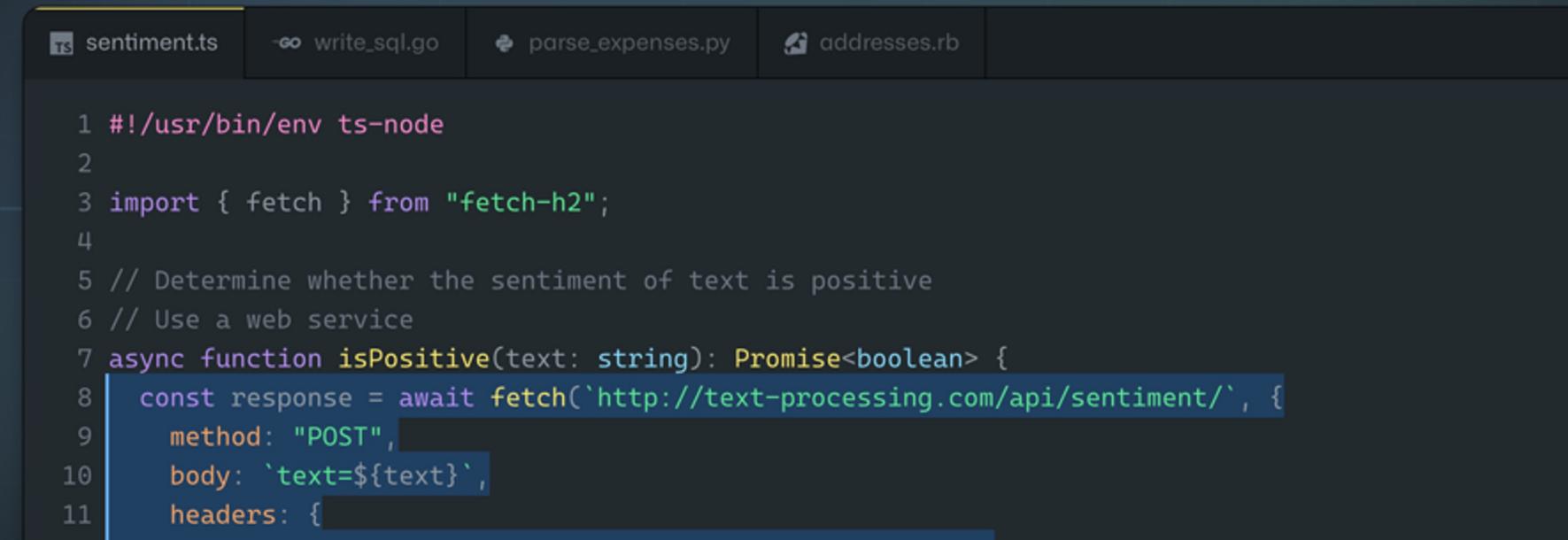
How does AlphaFold Server work?

AlphaFold Server is a web-service that can generate highly accurate biomolecular structure predictions containing proteins, DNA, RNA, ligands, ions, and also model chemical modifications for proteins and nucleic acids in one platform. It's powered by the newest AlphaFold 3 model.

Technical Preview

Your AI pair programmer

With GitHub Copilot, get suggestions for whole lines or entire functions right inside your editor.

[Sign up >](#)

A screenshot of a code editor interface. At the top, there are tabs for 'sentiment.ts', 'write_sql.go', 'parse_expenses.py', and 'addresses.rb'. The 'sentiment.ts' tab is active. Below the tabs, a snippet of TypeScript code is shown:

```
1 #!/usr/bin/env ts-node
2
3 import { fetch } from "fetch-h2";
4
5 // Determine whether the sentiment of text is positive
6 // Use a web service
7 async function isPositive(text: string): Promise<boolean> {
8   const response = await fetch(`http://text-processing.com/api/sentiment/`, {
9     method: "POST",
10    body: `text=${text}`,
11    headers: {
```

The code editor has a light blue selection bar highlighting the line 'const response = await fetch(`http://text-processing.com/api/sentiment/`, {'. This indicates that GitHub Copilot is currently suggesting or completing this line of code.



These Technologies Bring New Problems

NBC NEWS Lawsuit claims Character.AI is responsible for teen's suicide

ARTIFICIAL INTELLIGENCE

Lawsuit claims Character.AI is responsible for teen's suicide

Megan Garcia says the company's chatbots encouraged her 14-year-old son, Sewell Setzer, to take his own life, according to the lawsuit.

ARTIFICIAL INTELLIGENCE

How generative AI is boosting the spread of disinformation and propaganda

In a new report, Freedom House documents the ways governments are now using the tech to amplify censorship.

By Tate Ryan-Mosley

CES 2025 AWARDS

Chegg Is On Its Last Legs After ChatGPT Sent Its Stock Down 99%

With subscriptions tumbling, there are doubts the online education company will be able to pay its debts.

By Thomas Maxwell Published November 9, 2024 | Comments (75)

BEST OF CES 2025 AWARDS

THE BEST OF CES 2025 AWARDS

Best of CES 2025 Awards →

© Gordon Burtch, 2025

Business Process / Model Still Matters

Bloomberg Opinion

Money Stuff

Sorry, Zillow's Computer Can't Buy Your House Right Now

Also CEO pay, the Boredom Markets Hypothesis and Big Short guys being big short.

By Matt Levine [+Sign Up](#)
October 18, 2021, 1:18 PM EDT

Zillow

Matt Levine is a Bloomberg Opinion columnist covering finance. He was an editor of Dealbreaker, an investment banker at Goldman Sachs, a mergers and acquisitions lawyer at Wachtell, Lipton, Rosen & Katz, and a clerk for the U.S. Court of Appeals for the 3rd Circuit.

[Read more opinion](#)

Deciding how much you should pay for a share of large-cap publicly traded stock is not an *entirely* solved problem, but it's pretty close. If someone comes to you and says "hey I have 100 shares of Microsoft Corp. stock for sale, how much will you pay me for it," a pretty decent answer would be to look at the last price at which Microsoft traded – like a millisecond ago – and subtract, you know, one cent from that price. That will get you a price that is likely to be competitive (the seller might actually sell to you), likely to be profitable (you might be able to sell it for more than you paid), and

LIVE ON BLOOMBERG
Watch Live TV >
Listen to Live Radio >



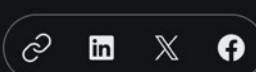
// Menu > Institutional > Tools About Archive Events

May 4, 2023

Google “We Have No Moat, And Neither Does OpenAI” // Leaked Internal Google Document Claims Open Source AI Will Outcompete Google and OpenAI

10 minutes
11 comments

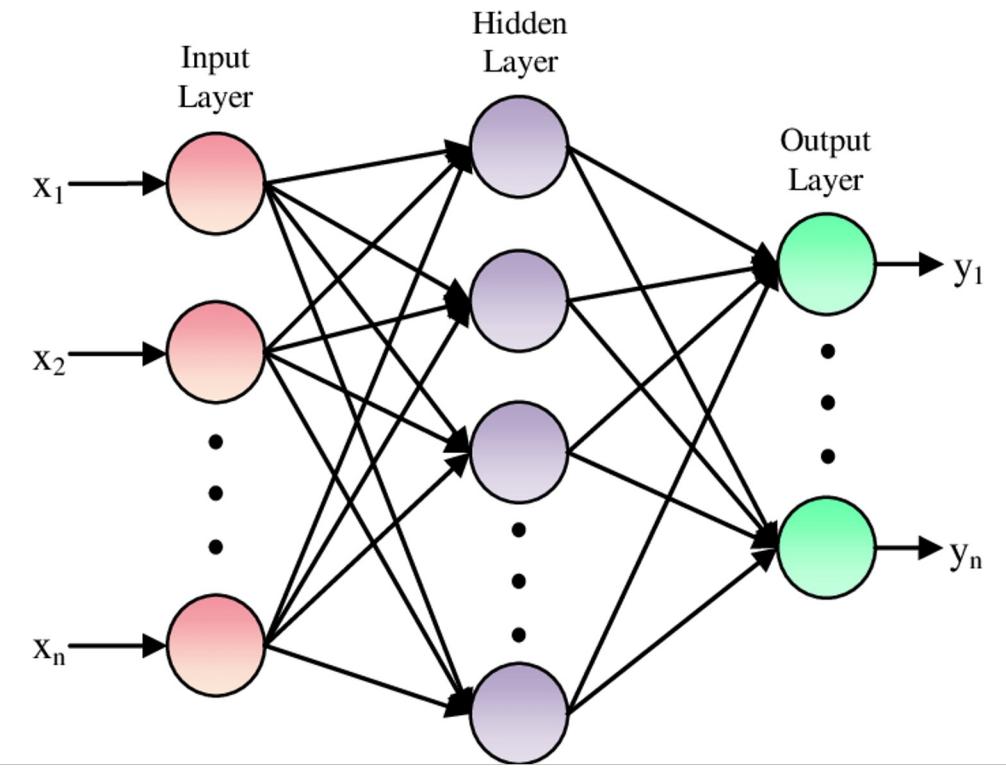
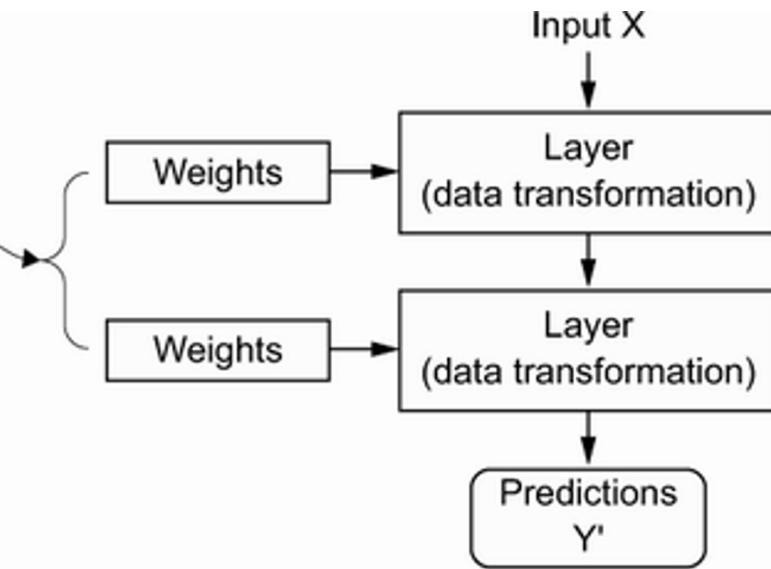
By Dylan Patel and Afzal Ahmad



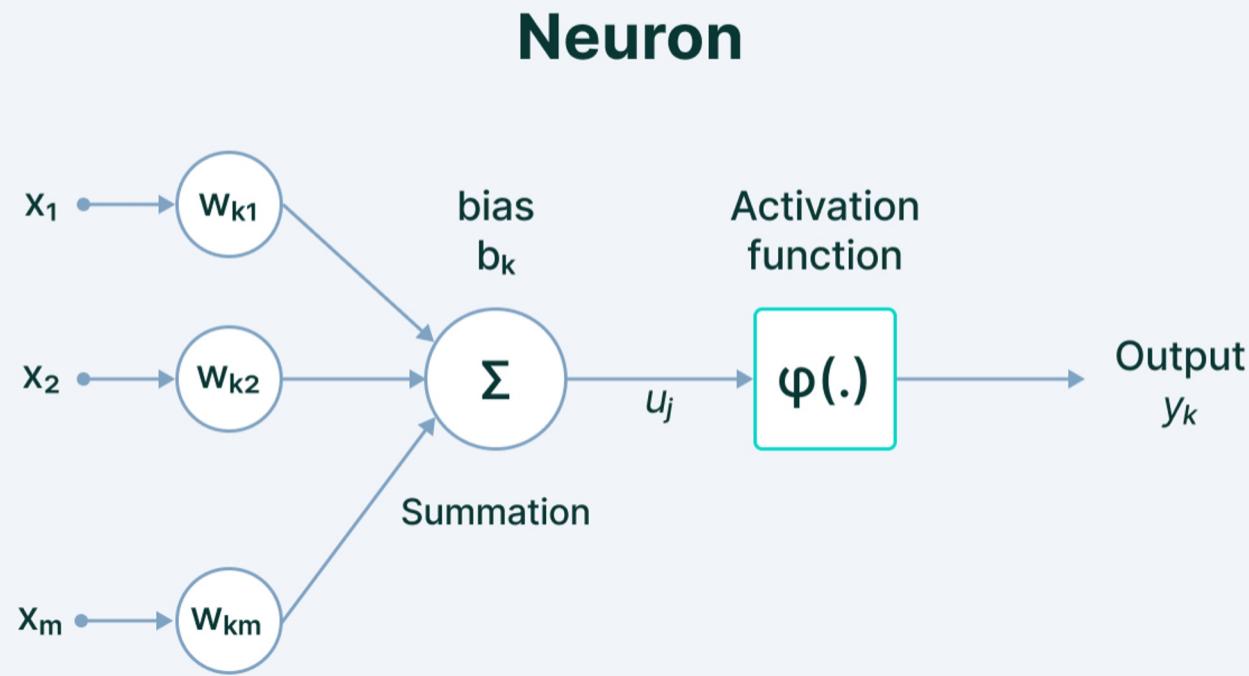
How It Works, Conceptually

Model Parameters

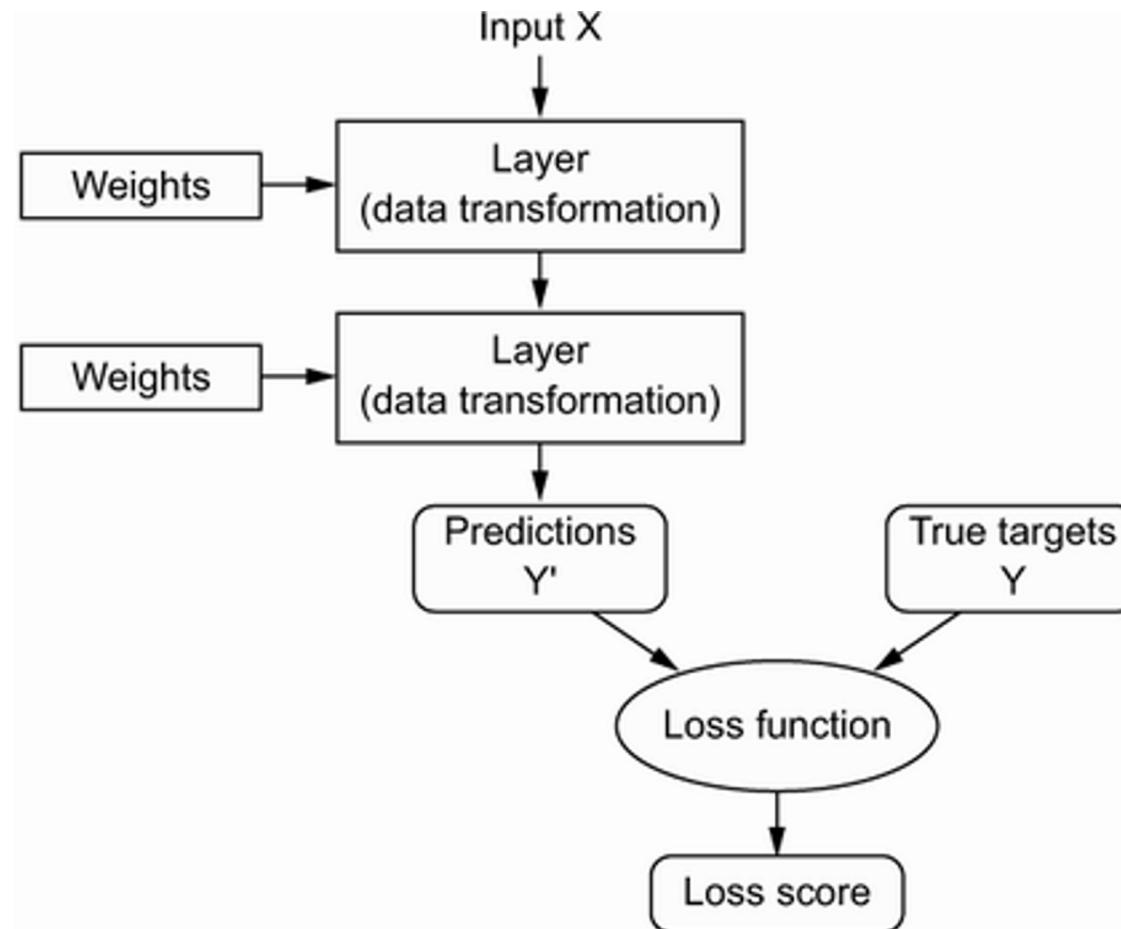
Goal: finding the right values for these weights



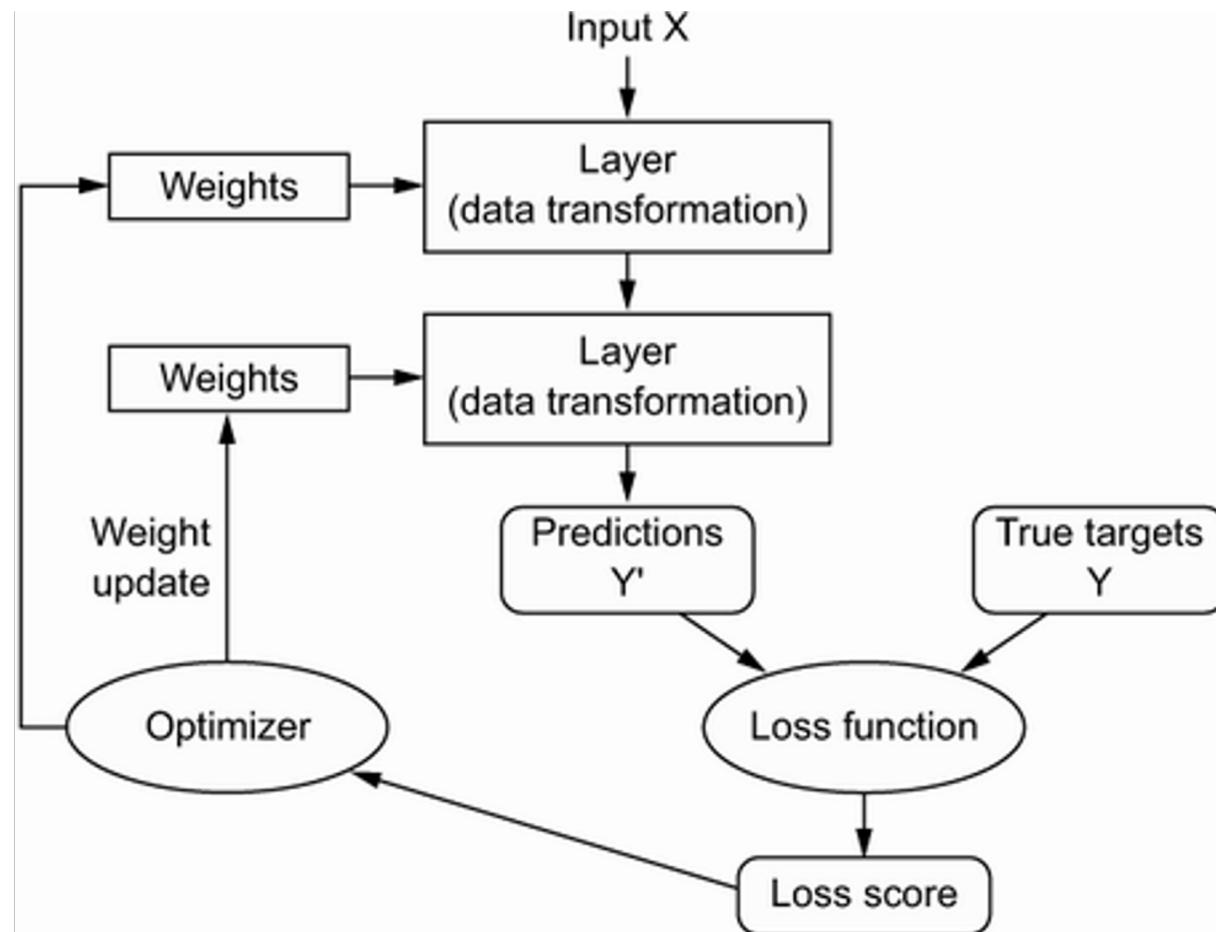
Model Parameters



Loss Function (Error)



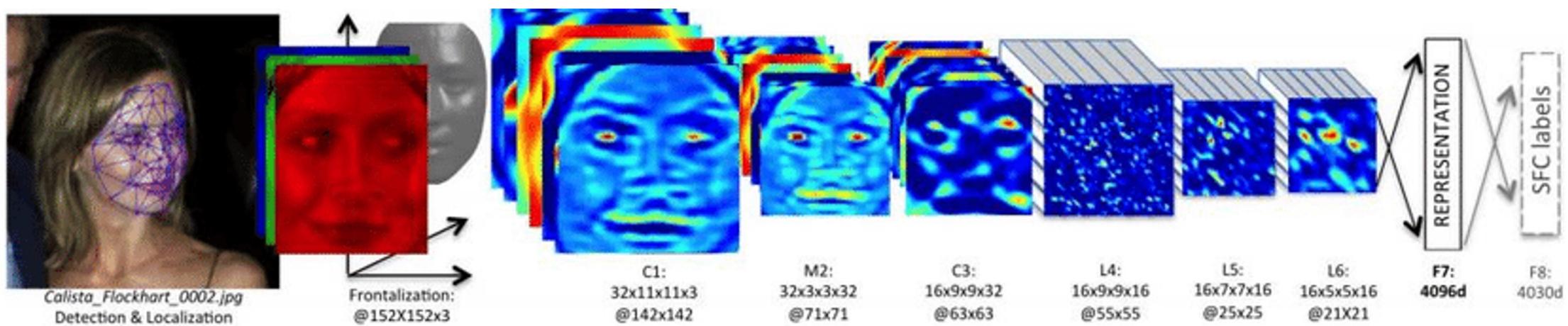
Optimization



When to Learn Deeply (vs. Not)

COMPLEX RELATIONSHIPS

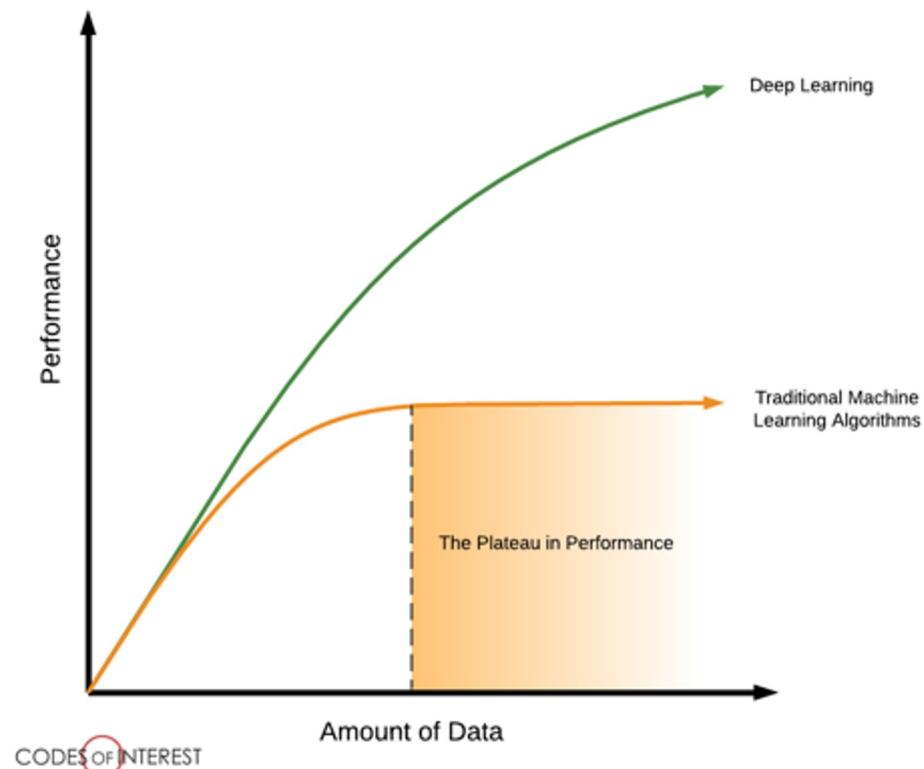
- Complex, non-linear, interactive relationships and mappings; common use cases involve unstructured (high dimensional) data. Deep learning techniques remove the need for feature engineering, a daunting task.



When to Learn Deeply (vs. Not)

LOTS OF DATA ON HAND

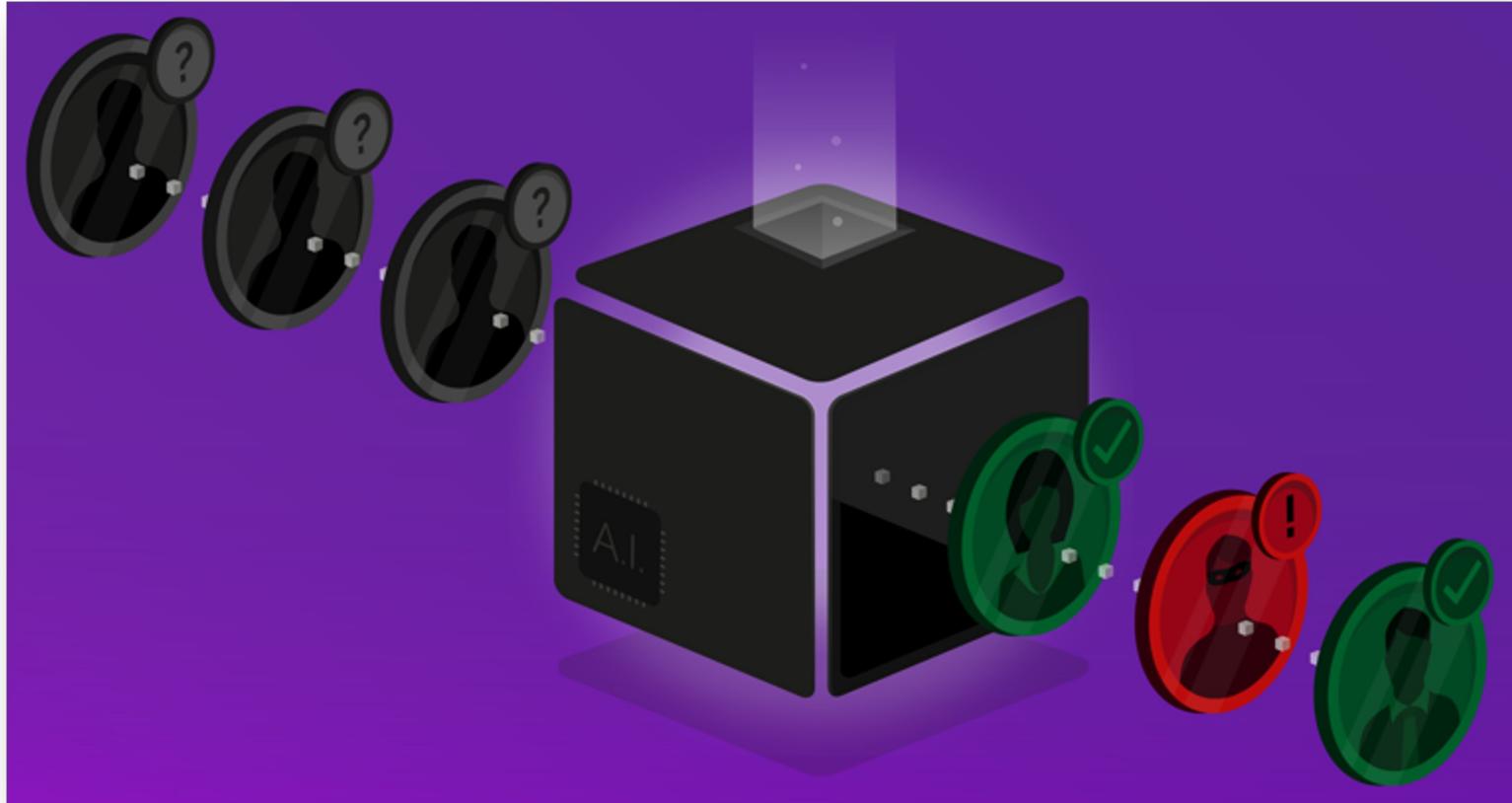
- To be able to learn those complex mappings, typically requires many, many, many training examples.



When to Learn Deeply (vs. Not)

LITTLE NEED FOR UNDERSTANDING

- Although there have been advancements in explainable and interpretable AI, deep nets are notoriously “black box” algorithms.

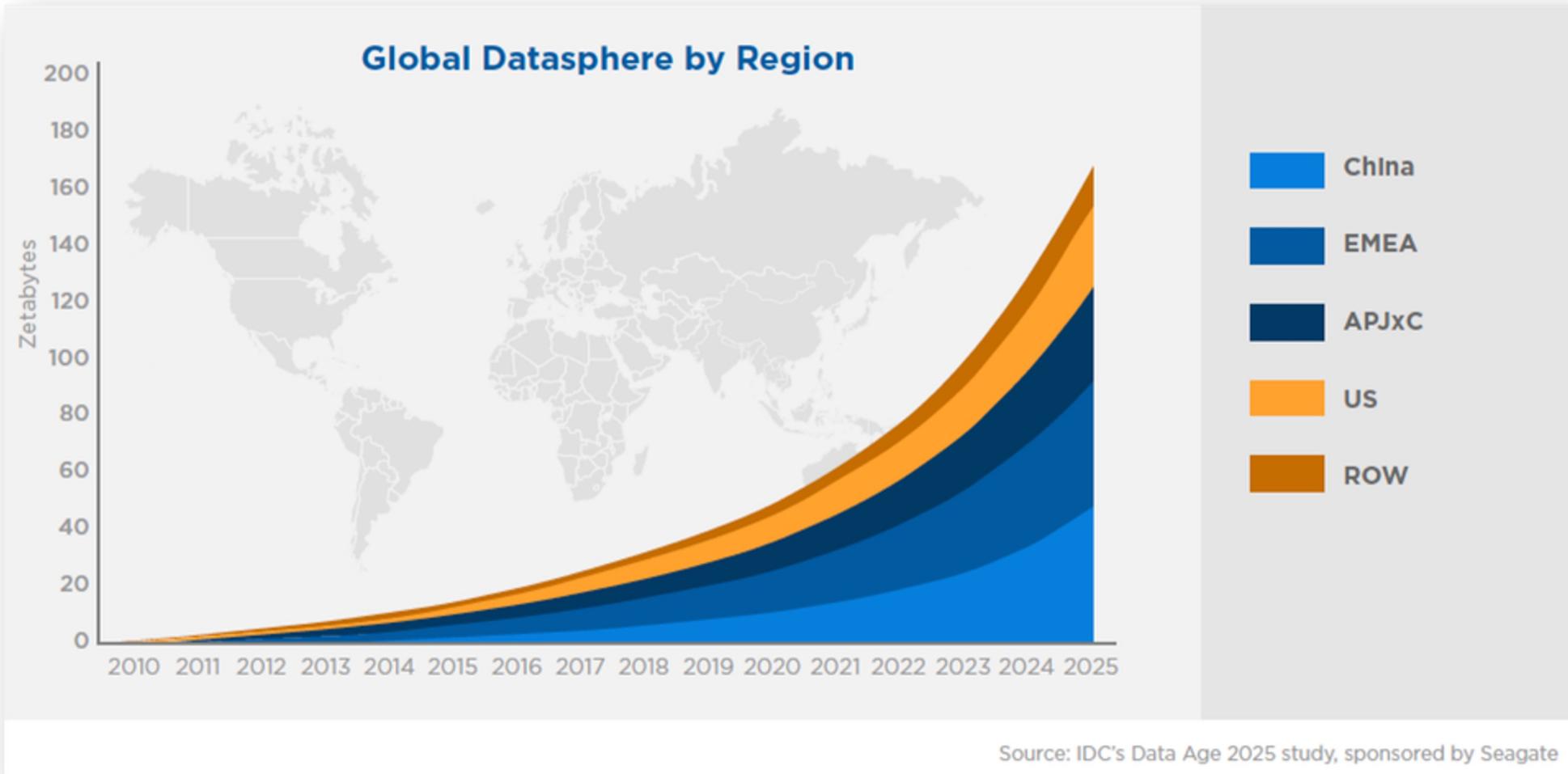


Why Did Deep Learning Take Off?

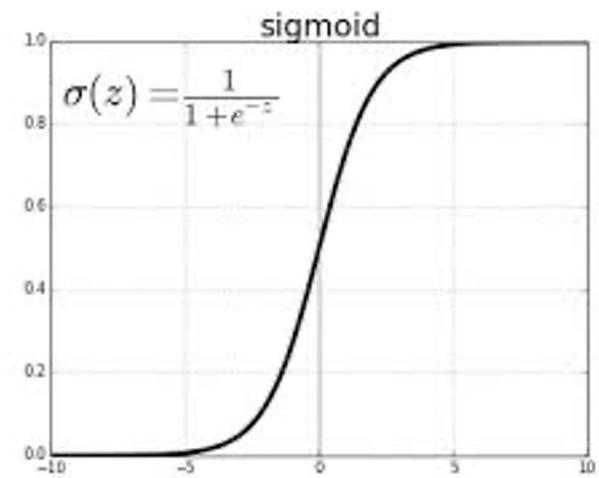
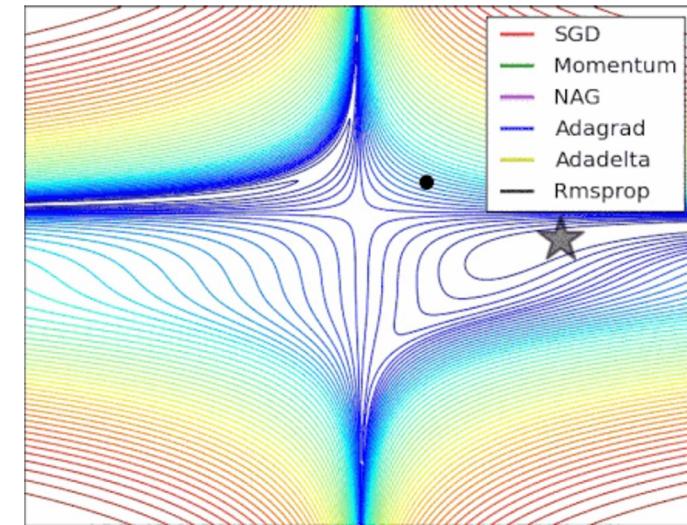
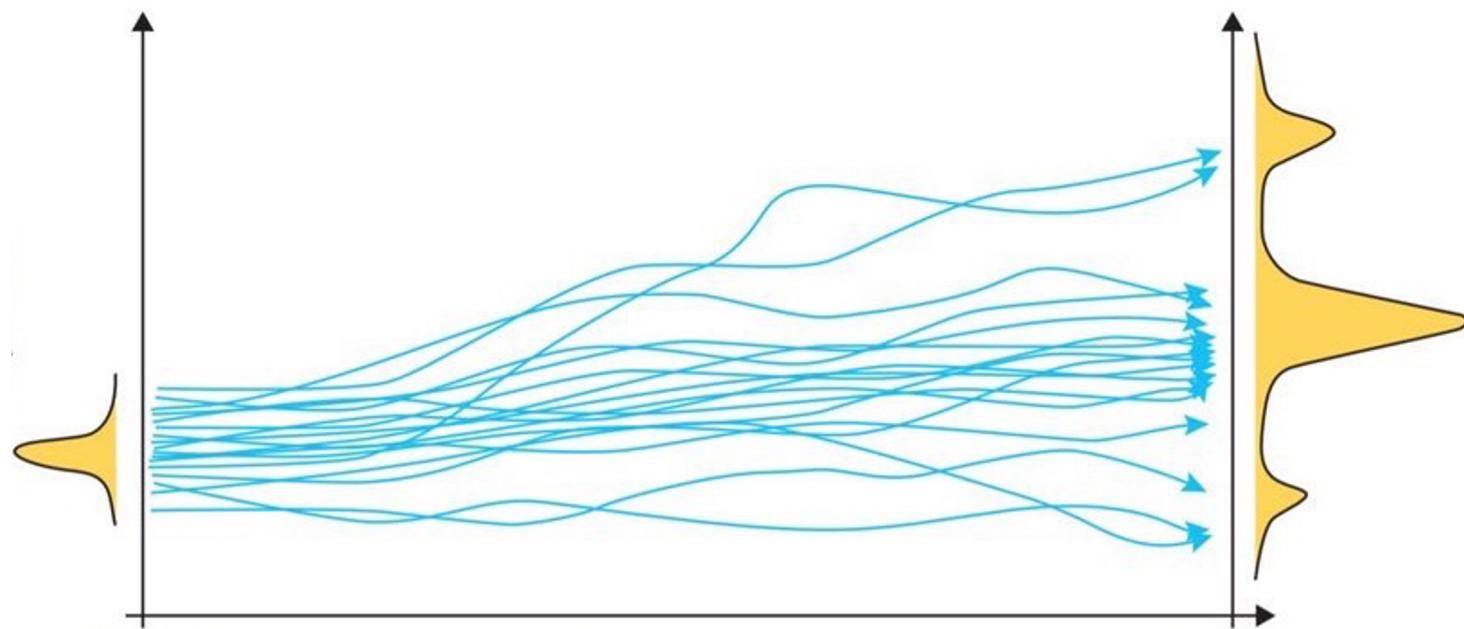


Video Games

Data



Algorithmic Improvements



Questions?