Contents

Interpretability in Artificial Intelligence

SOUL course

Course Description

As artificial intelligence models like chatGPT become increasingly capable and ubiquitous, the need to understand their inner workings intensifies. Imagine an autonomous vehicle taking an unexpected turn or a medical AI diagnosing a life-altering condition; the importance cannot be overstated. Despite their widespread applications, our grasp of these models remains alarmingly limited. This course is designed to survey this gap. A special emphasis is placed on *mechanistic interpretability*, a subfield that rigorously investigates AI networks at the neuronal level.
The course will engage students through rigorous reading assignments, interactive discussions, and a hands-on project. Papers from researchers/groups in this area, such as AnthropicAI, OpenAI, and MIT's Tegmark Group, form the cornerstone of our course.

Course Topics**

This was the original one, and maybe completely modified accordingly to student desire. Various research papers and articles to be distributed during the course.

1. Introduction to Interpretability in AI [1 week] and basics of transformer language models. [1 Week]

2. Mechanistic Interpretability [4 weeks], example papers include, [19], and a good collection of articles is collected here. One over arching goal is to reproduce the paper [18], which shows the algorithms in the phenomena of gorkking uses discrete Fourier transforms and trigonometric identities to convert addition to rotation about a circle.

3. Concept-based Interpretability [1 week] example papers include, [5].

Required Background

Students are expected to have a basic understanding of calculus, linear algebra, probability theory, and coding. A reading list will be provided to be completed before the commencement of the course.

Assessment

Student assessment will be based on both weekly reading assignments and a course-long project. Students can opt for either a coding project or a written project, aimed at deeply exploring a sub-field of interpretability. The grade distribution will be as follows: Weekly Readings: 60%, Course-long Project: 40%.

1 Timeline

Week 1: introduction to Alignment

Introduction to the topics and scope of the course. The goal is to give a sense of the topics involved in the AI Safety community where the topics of the course fit in this picture. As examples, we discuss the problems of alignment and robustness in a nontechnical way. Some of these concerns are near-term: how do we prevent driverless cars from misidentifying a stop sign in a blizzard? Others are more long-term: if general AI systems are built, how do we make sure these systems pursue safe goals and benefit humanity?

Textbook and references: The field is extremely young, with no complete surveys about this. However, there are a number of useful survey article on this subject.

- The main article we will be looking at is the survey *Unsolved Problems in ML safety*, [10]. This is by Dan Hendricks at the center of AI safety.

- For broader perspective, there are articles by Rohin Shah [1] and Nanda [17].

Homework: this week's homework will require student to read over certain articles that examinines the socio-technical complexities of ensuring that AI and humans share compatible goals.

Week 2: The mathematical prerequisites

The language of matrices will form the basis of our discussion, we will go through a few basic matrix manipulation. The background is minimal, we do not require full Linear Algebra course background equivalent to that of 110.201, but similar content to the first 2 weeks.
Textbook and references:

- Linear Algebra with Applications, 5th Edition, [4], Otto Bretscher, Prentice Hall, December 2012, ISBN-13: 978-0321796974. This is the course text book for 110.201.
- A concise and sufficient introduction is in Stanford CS229, see [16, Sec. 1-3].
- Section 1-3 of Strictland's notes lectures [22, p1-3]

Week 3: Learning Methods and Robustness

The goal of this week is to give an overview of what is meant by *AI*, in the context of this course. This involves introducing the the first of the three most common machine learning techniques;

1. Supervised learning.
2. Reinforcement learning.
3. Unsupervised learning.

We will discuss the second technique the following week, and not really go in detail of the third example.

The examples is used to understand the *capabilities* of ML method, and highlight examples of failure modes, particularly in the context of adversarial examples, [13], see **??** for a more detailed collection of references.

One important aspect of this week is that we will phrase the machine learning paradigm in terms of *function learning*. This is particularly simplistic but is sufficient for our purpose.
References and texts:

- This lecture will be a summary of the field of ML, similar to Ngo's post.
- The requisite is minimal and we will be going through the most basic supervised learning example.

Week 4: Homework discussion and buffer

Week 5 - 6: Reinforcement Learning and Goal Misgeneralization

We will discuss one-two basic examples of each, and finish with how this leads to the problem of outer and inner alignment. The main reference for this week is by Rohin Shah et al. on *Goal Misgenralizations*, [20]. As in previous topics we will go through the main example in the paper, and discuss current methods of approaching it.
References:

- We will cover section 1 of the reinforcement learning textbook. We will use the classic reference by Sutton and Barto [23].

Weeks 7: Oversight of AI Systems

Exploration of the challenges in AI system oversight. Topics include reinforcement learning from human feedback [7]. Introductory talk [14].

Week 8: Homework discussion and buffer

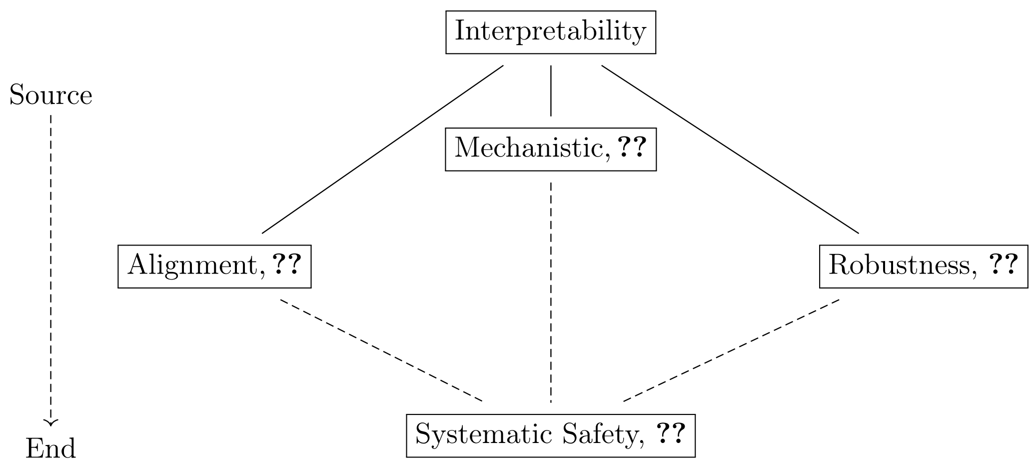
2 Safety in AI

References:
At a high level overview definition

Definition 2.1. *Machine learning* methods are automatic methods for extracting information from data for various *AI tasks.*

2.1 A bird eye view of the subject

To organize the article, I categorize the various research areas.



The vertical hierarchy can be thought of as getting from the source (foundational design of the AI systems) to end (their use and place within society). The arrows only mean "contribute". A brief summary of each box is given in, 2.2. Importantly, the whole lifecycle of AI has various stakeholders involved:

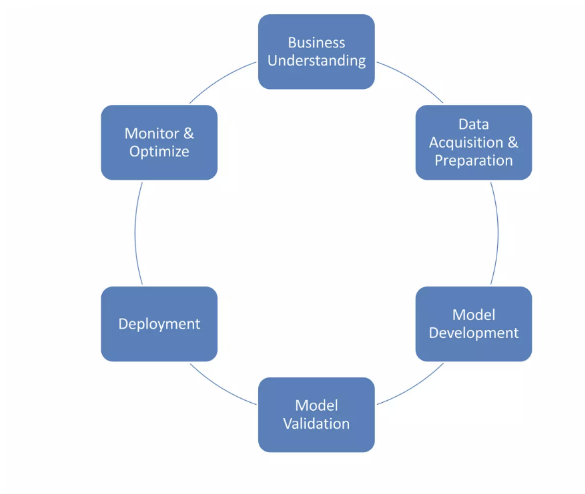


Figure 1: The AI Life Cycle

We refer to [8] for further details.

2.2 Overview of each field

We will focus on three of four types of problems, as explained in [10].

- *Robustness.* There are broadly two types of events:
  - *Black swan*: rare extreme, unsual events. Examples of such include the 2010 Flash Crash, [15], or those appearing in autonomous vehicle. In the context of statistics, these are also referred to as *long tail events.* Long tail distributions make the usual arsenal of statistical tools can become less useful. Examples include power law distributions.
  - *Adversaries*: carefully crafted threats, or as Goodfellow et al. describes, " *adversarial examples* are inputs to machine that an attacher has intentionally designed to casue the model to make a mistake." Much literature has focused on "perturbation defense", which was mostly focused from Szegedy et al's, [24]. We discuss more details of this in Sec. **??**.
- *Alignment*, we list a number of main difficulties in alignment.
  - specification. Encoding goals such as judgment, experiences, and happiness is hard. Research is done to assess the language model's ability to evaluate scenarios that could be morally contentions. [9].
  - even when one is able to describe a value to be learned:

- it can be difficult to optimize, as models can have unintended undesirable secondary objectives. To understand and steer agent's moral values, researchers could dconstruct relevant environments, for instance, Jiminy Cricket, which has 25 text-based games with diverse scenarios, [11].
- *brittle*, due to reasons as proxy gaming. Further, even with "human approved" proxies, there is the risk of deception from computers. Future systems should strive to address the problem of verification, past attempts include an adversarial process, where during training two agent takes turn making short statements with human judges on the final result, [12].

- *systematic safety*, is the deployment of ML systems in the larger context, including software systems, organizational structure or human society. Two main examples of systematic research are in cybersecurity, such as the defense of potential hackers, and informed decision making.

2.3 Homework

Homework for week 1 is a reading exercise.
In this week, you will read the following two papers, [10], [2]. For the first paper,

1. Describe a nontrivial strength or weakness of the paper that isn't explicitly mentioned in the readings themselves. (300-400)

2. Summarize the reading. Be sure to read and summarize the contents of each section; do not just describe the overall idea of the paper/article. (400-500)

For the second paper (which is more technical),

3. Write one concept in the article that confuses you/you hope to understand more. Explain why.

3 Linear Algebra Review

Learning Objectives

- Learn the basic language of linear algebra.

Notations:

- $\mathbb{R}$ denotes the real numbers.
- $\mathbb{Z}$ denotes the set of integers. We let $[n] = \{1, 2, \ldots, n\}$ denote the set of the first n positive integers.
- For an arbitrary set X, X^n will dentoe the set of ordered sequences of n elements in X. The set is also denoted as $\{(x_1, \ldots, x_n) : x_i \in X\}$.
- We assume the reader is familiar with the basic terminology of functions and sets.

Definition 3.1. A $m \times n$ matrix in $\mathbb{R}$ is a function

$$[n] \times [m] \to \mathbb{R}$$

This is often denoted in two ways:

- as a square with m-row and n-columns.
- or from the compact notation $(a_{ij})_{i \in [m], j \in [n]}$.

For example, below are 2×3

$$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix}$$

a 3×2 matrix

$$\begin{pmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{pmatrix}$$

Definition 3.2. Let $M_{m \times n}(\mathbb{R})$ be the set of $m \times n$ matrices.

4 Neural Networks

References: [21, 1-2].

Learning Objectives

- Understand what a neural network is. Understand its difference to classical statistics.
- Phrasing AI tasks in terms of functions.
- Play with the TensorFlow playground. [a]

[a] We will *not* need to know how to code.

Question

How can an AI system fail? A taxonomy via observability, [6]. Now that we know the pipeline of machine learning: there are two areas

1. *Observable failures*
2. *Unobservable failures*

We will begin by showing how many AI tasks such as classification be rephrased as learning a function.

$$\{\text{Machine learning problem}\} \longrightarrow \{\text{Represented as a function}\}$$

This language, especially in sec [transformers], reduces the verbosity of the presentation.

4.1 Overview of supervised learning

Consider classifying whether an image is a cat. Solving this problem is to find a function

$$f : \{0, 1, \dots, 255\}^{28\times 28} \to [0, 1]$$
$$f(\text{image}) = \begin{cases} 0 & \text{image is cat} \\ 1 & \text{image is not cat} \end{cases}$$

The domain of the function is the set of all possible images when expressed in vector form: an image has 28×28 pixels and a value of 0-255 in grayscale. This is often referred to as the PAC or SLT model.

Example

- Classification on whether an object is a chair or not.
- Detecting whether an email is spam or not. We can have a number of emails labeled (by human) according to a number of features.

"money"	"pills"	bad spelling	Known sender	spam?
Y	N	Y	N	Y
Y	Y	Y	Y	N
...	...	...	...	...

  - There are four features in this table. This will often be depicted as a vector, to encode the "Y=Yes" or "N=No".

Our goal is to approximate a function:

$$f : \mathbb{R}^4 \to \mathbb{R}$$

$$f(\text{features of email}) = \begin{cases} 1 & \text{if spam} \\ 0 & \text{if not spam} \end{cases}$$

where the data can be represented in terms of a vectors in $\mathbb{R}^4$. We may also use the first two features as training, to approximate a function:

$$f : \mathbb{R}^2 \to \mathbb{R}$$

Definition 4.1. A *supervised learning method* is a method that creates a function (*predictor*) that *generalizes* well from *input data.*

The typical set up can be visualized in the following diagram:

$$\begin{array}{c} \text{Data (with a distribution } \mathcal{D}) \\ \downarrow{\scriptstyle (x_1,\ldots,x_n)} \\ \text{Human Expert} \\ \downarrow{\scriptstyle \{(x_1,y_1),\ldots,(x_n,y_n)\}} \\ \text{Algorithm} \\ \downarrow \\ \text{model} \xleftrightarrow{\text{compare, } \mathbb{E}(l(f,t))} \text{reality} \end{array}$$

Input: - An *assumed* true function $t : \mathbb{R}^n \to \mathbb{R}$.

- Finite set of data sampled from real world :

$$(x_i, y_i)_{i=1}^N := \{(x_1, y_1), \ldots, (x_N, y_N)\}$$

  This is the *ground truth*.

Output: Some function $f : \mathbb{R}^n \to \mathbb{R}$ it *generalizes*, 4.2, well to the true function

The three theoretical aspects, each of intensive study are thus:

1. Optimization. How does it behave? Why does optimization work so well?
2. Approximation error. This is the choice of the architecture, or the its *expressitivity*.
3. Generalization. There is where we see the phenomena of double descent, [3].

4.2 What it means to generalize

In an ideal world, where we have access to a true function, we can measure the discrepancy between our predictor and true function using

Definition 4.2. The *true risk* of a predictor, f, is

$$t\text{-err}(f) := \mathbb{E}[l(f, t)]$$

where

$$l(f, t) : X \to \mathbb{R}$$

is a choice of loss function and $t : X \to \mathbb{R}$ is the *true* function.

Definition 4.3. The *bext predictor* is an element,

$$f^* \in \arg\min_f t\text{-err}(f)$$

Definition 4.4. The 0-1 loss function.

$$l(f, t)(x) = 1_{f \neq t}(x) = \begin{cases} 0 & \text{if } f(x) \neq t(x) \\ 1 & \text{if } f(x) = t(x) \end{cases}$$

Example

If $\mathcal{D}$ is the uniform distribution on[a] $X \subseteq \mathbb{R}^n$,

$$\mathbb{E}(l_{0,1}(f, t)) = |\{x \,:\, f(x) \neq t(x)\}|$$

where $| \quad |$ is the area[b] of the set.

[a] To be precise, a Borel measurable set.
[b] more precisely, Borel measure

Definition 4.5. The *training error* or the *empirical risk* of a function on a sample data

$$S := \{(x_i, t(x_i) =: y_i\}$$

$$\mathrm{er}_S(f) := \mathbb{E}_S(l_{0,1}(f,t)) = \frac{|\{x_i : f(x_i) \neq y_i\}|}{N}$$

The *minimal emprical risk* is the set

$$\mathrm{ERM}_S(f) := \arg\min_f \mathrm{er}_S(f)$$

Does minimizing empirical risk imply minimizing true risk?

Example

Let us suppose $\mathcal{D}$ is the uniform distribution. We are to approximate the function

$$t : [0,1]^2 := \{(a,b) : a,b \in [0,1]\} \to [0,1]$$

using only a finite set, $S = \{(x_i, y_i)\}$, of ground truths. Suppose on S, we define

$$f_S(x) = \begin{cases} y & \text{if } x \in S \\ 0 & \text{otherwise} \end{cases}$$

Then

$$\mathrm{er}_S(f_S) = 0$$

In other words,

$$f_S \in \arg\min_f \mathrm{er}_S(f)$$

However

$$t\text{-err}(f_S) = 1/2$$

This is the problem of *overfitting*.

Question

What kind of functions $\mathbb{R}^n \to \mathbb{R}$ do we get from feed forward networks with one hidden layer? (Depends on σ and width).

One can understand the typical pipeline in the following Colab note book.

4.3 Function space

4.4 Linear learning

Example

Can't learn XOR or the circle function.

4.5 Kernel methods

$$\mathbb{R}^n \xrightarrow{\text{non-linear}} \mathbb{R}^N \longrightarrow \mathbb{R}$$

The non-linear map is often called the *feature map*. Many tasks in traditional (pre-Neural network) is addressed using this.
Another example is : SVM.

$$\{\text{Blair, Bush}\} \xrightarrow{\text{PCA}} \mathbb{R}^{150} \longrightarrow \mathbb{R}$$

4.6 Neural networks

What we learn is complex, how we learn it might be simple. - paraphrasing Geoffrey Hinton. The brain has approximately 80 billion neurons.

Definition 4.6. A *perceptron layer* is the datum of:

$$p(W, b, \sigma) : \mathbb{R}^n \xrightarrow{A_{W,b}} \mathbb{R}^m \xrightarrow{\sigma} \mathbb{R}^m$$

where

$$A_{W,b}(x) := Wx + b, W \in \mathbb{R}^{m\times n}, b \in \mathbb{R}^{m\times 1}$$

is called an *affine transormation.* σ is a *any* function.

Choices of σ include

- ReLU $:= \max(0, -) : \mathbb{R} \to \mathbb{R}$ function. This induces component wise of the function ReLU $: \mathbb{R}^n \to \mathbb{R}$.
- You would *not* want σ to be linear. The nonlinearity depends on a choice of basis. This destroys the symmetry[1], increasing in expressivity.
- Historically, the functions $\sigma(x \cdot w + b)$ are referred as *ridge functions.*

Definition 4.7. A *(vanilla) multilayer perception* or *nerual net* consists of a composition of multilayer percetpron

$$\mathbb{R}^{n_1} \xrightarrow{p_1} \mathbb{R}^{n_2} \xrightarrow{p_2} \cdots \longrightarrow \mathbb{R}^{n_k}$$

[1] If one is allowed to change the basis, then we can simply find a new basis so that all the functions are well behaved.

4.6.1 Choice of loss function

4.7 Gradient descent

5 Homework

Homework for week 2 is reading exercise, Due Wednesday.
In this week: choose *one* of the following options (1 or 2): [2]

1. For those still want to familiarize themselves with supervised learning:
   - Read Vishal Maini. Answer (c).
   - Read a visualization on the progress and a survey by Ngo. Answer (a),(b) and (c) for the second paper.
2. For those who are already familiar with deep learning,
   - Read Zoom In. Answer (a), (b) and (c).
   - Read the case study of analyzing CNN. Answer (a) and (b).

General questions:

(a) Describe a nontrivial strength or weakness of the paper that isn't explicitly mentioned in the readings themselves. (300-400)

(b) Summarize the reading. Be sure to read and summarize the contents of each section; do not just describe the overall idea of the paper/article. (400-500)

(c) Write one concept or technicality in the article that confuses you/you hope to understand more. Explain why.

[2] the alphabet letters refer to the "general questions"

References

[1] *AI Alignment 2018-19 Review — AI Alignment Forum — alignmentforum.org.* [Accessed 28-09-2023].

[2] Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané, *Concrete problems in ai safety*, 2016.

[3] Mikhail Belkin, *Fit without fear: remarkable mathematical phenomena of deep learning through the prism of interpolation*, 2021.

[4] Otto Bretscher, *Linear algebra with applications, 5th edition*, 2012.

[5] Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt, *Discovering latent knowledge in language models without supervision*, 2022.

[6] Stephen Casper, *Eight Strategies for Tackling the Hard Part of the Alignment Problem — AI Alignment Forum — alignmentforum.org.* [Accessed 08-11-2023].

[7] Paul Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei, *Deep reinforcement learning from human preferences*, 2023.

[8] Marina Danilevsky, Shipi Dhanorkar, Yunyao Li, Lucian Popa, Kun Qian, and Anbang Xu, *Explainability for natural language processing*, Proceedings of the 27th acm sigkdd conference on knowledge discovery & data mining, 2021, pp. 4033–4034.

[9] Dan Hendrycks, Collin Burns, Steven Basart, Andrew Critch, Jerry Li, Dawn Song, and Jacob Steinhardt, *Aligning ai with shared human values*, 2023.

[10] Dan Hendrycks, Nicholas Carlini, John Schulman, and Jacob Steinhardt, *Unsolved problems in ml safety*, 2022.

[11] Dan Hendrycks, Mantas Mazeika, Andy Zou, Sahil Patel, Christine Zhu, Jesus Navarro, Dawn Song, Bo Li, and Jacob Steinhardt, *What would jiminy cricket do? towards agents that behave morally*, 2022.

[12] Geoffrey Irving, Paul Christiano, and Dario Amodei, *Ai safety via debate* (2018).

[13] Robin Jia and Percy Liang, *Adversarial examples for evaluating reading comprehension systems*, 2017.

[14] Jared Kaplan, *AI Safety, RLHF, and Self-Supervision - Jared Kaplan | Stanford MLSys, youtube.com.* [Accessed 28-09-2023].

[15] Andrei Kirilenko, Albert S. Kyle, Mehrdad Samadi, and Tugkan Tuzun, *The flash crash: High-frequency trading in an electronic market*, The Journal of Finance **72** (2017), no. 3, 967–998.

[16] Zico Kolter and Chuong Do, 2015.

[17] Neel Nanda, *My Overview of the AI Alignment Landscape: Full Sequence — docs.google.com.*

[18] Neel Nanda, Lawrence Chan, Tom Lieberum, Jess Smith, and Jacob Steinhardt, *Progress measures for grokking via mechanistic interpretability*, 2023.

[19] Chris Olah, Nick Cammarata, Ludwig Schubert, Gabriel Goh, Michael Petrov, and Shan Carter, *Zoom in: An introduction to circuits*, Distill (2020). https://distill.pub/2020/circuits/zoom-in.

[20] Rohin Shah, Vikrant Varma, Ramana Kumar, Mary Phuong, Victoria Krakovna, Jonathan Uesato, and Zac Kenton, *Goal misgeneralization: Why correct specifications aren't enough for correct goals*, 2022.

[21] Shai Shalev-Shwartz and Shai Ben-David, *Understanding machine learning - from theory to algorithms*, 2014.

[22] Neil Strickland, *Linear mathematics for applications*, 2020.

[23] Richard S. Sutton and Andrew G. Barto, *Reinforcement learning: An introduction*, IEEE Transactions on Neural Networks **16** (2005), 285–286.

[24] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, D. Erhan, Ian J. Goodfellow, and Rob Fergus, *Intriguing properties of neural networks*, CoRR **abs/1312.6199** (2013).