Artificial Intelligence and Machine Learning

Adriana Fidanza

April 26, 2023

Contents

1 Introduction

Artificial Intelligence (AI) and Machine Learning (ML) are rapidly growing fields in computer science that have significant implications for how we live and work.

Over the past few years, the concept of AI and machine learning has captured the attention of the world. It has taken the world by storm, and the application of machine learning is being observed in every walk of life. But, what are AI and machine learning? How do they work? And why are they so important? I hope to answer these questions and more in my speech today.

In this paper, we will explore the key concepts and applications of AI and ML, their historical development, and the challenges and opportunities that they present.

2 Overview of AI and ML

You might hear people use artificial intelligence (AI) and machine learning (ML) interchangeably, especially when discussing big data, predictive analytics, and other digital transformation topics. The confusion is understandable as artificial intelligence and machine learning are closely related. However, these trending technologies differ in several ways, including scope, applications, and more. So, what exactly is the difference when it comes to ML vs. AI, how are ML and AI connected, and what do these terms mean in practice? We'll break down AI vs. ML and explore how these two innovative concepts are related and what makes them different from each other.

2.1 What is Artificial Intelligence?

AI refers to the ability of machines to perform tasks that would normally require human intelligence, such as recognizing patterns, making decisions, and learning from experience. Artificial intelligence is a broad field, which refers to the use of technologies to build machines and computers that have the ability to mimic cognitive functions associated with human intelligence, such as being able to see, understand, and respond to spoken or written language, analyze data, make recommendations, and more. Although artificial intelligence is often thought of as a system in itself, it is a set of technologies implemented in a system to enable it to reason, learn, and act to solve a complex problem.

2.2 What is Machine Learning?

ML, on the other hand, is a subset of AI that focuses on creating algorithms that allow machines to learn from data and improve their performance over time. Machine learning is a subset of artificial intelligence that automatically enables a machine or system to learn and improve from experience. Instead of explicit programming, machine learning uses algorithms to analyze large amounts of data, learn from the insights, and then make informed decisions. Machine learning algorithms improve performance over time as they are trained—exposed to more data. Machine learning models are the output, or what the program learns from running an algorithm on training data. The more data used, the better the model will get.

2.3 How are AI and ML connected?

While AI and ML are not quite the same thing, they are closely connected. The simplest way to understand how AI and ML relate to each other is:

- AI is **the broader concept** of enabling a machine or system to sense, reason, act, or adapt like a human
- ML is **an application** of AI that allows machines to extract knowledge from data and learn from it autonomously

One helpful way to remember the difference between machine learning and artificial intelligence is to imagine them as umbrella categories. Artificial intelligence is the overarching term that covers a wide variety of specific approaches and algorithms. Machine learning sits under that umbrella, but so do other major sub-fields, such as deep learning, robotics, expert systems, and natural language processing.

2.4 Differences between AI and ML

While artificial intelligence encompasses the idea of a machine that can mimic human intelligence, machine learning does not. Machine learning aims to teach a machine how to perform a specific task and provide accurate results by identifying patterns.

Let's say you ask your Google Nest device, "How long is my commute today?" In this case, you ask a machine a question and receive an answer about the estimated time it will take you to drive to your office. Here, the overall goal is for the device to perform a task successfully—a task that you would generally have to do yourself in a real-world environment (for example, research your commute time).

In the context of this example, the goal of using ML in the overall system is not to enable it to perform a task. For instance, you might train algorithms to analyze live transit and traffic data to forecast the volume and density of traffic flow. However, the scope is limited to identifying patterns, how accurate the prediction was, and learning from the data to maximize performance for that specific task.

Artificial intelligence

- AI allows a machine to simulate human intelligence to solve problems
- The goal is to develop an intelligent system that can perform complex tasks
- We build systems that can solve complex tasks like a human
- AI has a wide scope of applications
- AI uses technologies in a system so that it mimics human decision-making
- AI works with all types of data: structured, semi-structured, and unstructured
- AI systems use logic and decision trees to learn, reason, and self-correct

Machine learning

- ML allows a machine to learn autonomously from past data
- The goal is to build machines that can learn from data to increase the accuracy of the output
- We train machines with data to perform specific tasks and deliver accurate results
- Machine learning has a limited scope of applications
- ML uses self-learning algorithms to produce predictive models
- ML can only use structured and semi-structured data
- ML systems rely on statistical models to learn and can self-correct when provided with new data

AI and ML bring powerful benefits to organizations of all shapes and sizes, with new possibilities constantly emerging. In particular, as the amount of data grows in size and complexity, automated and intelligent systems are becoming vital to helping companies automate tasks, unlock value, and generate actionable insights to achieve better outcomes.

Here are some of the business benefits of using artificial intelligence and machine learning:

- **Wider data ranges** Analyzing and activating a wider range of unstructured and structured data sources.
- **Faster decision-making** Improving data integrity, accelerating data processing, and reducing human error for more informed, faster decision-making.
- **Efficiency** Increasing operational efficiency and reducing costs.
- **Analytic integration** Empowering employees by integrating predictive analytics and insights into business reporting and applications.

2.5 Applications of AI and ML

Artificial intelligence and machine learning can be applied in many ways, allowing organizations to automate repetitive or manual processes that help drive informed decision-making. Companies across industries are using AI and ML in various ways to transform how they work and do business. Incorporating AI and ML capabilities into their strategies and systems helps organizations rethink how they use their data and available resources, drive productivity and efficiency, enhance data-driven decision-making through predictive analytics, and improve customer and employee experiences. Here are some of the most common applications of AI and ML:

- **Healthcare** and life sciences Patient health record analysis and insights, outcome forecasting and modeling, accelerated drug development, augmented diagnostics, patient monitoring, and information extraction from clinical notes.
- **Manufacturing** Production machine monitoring, predictive maintenance, IoT analytics, and operational efficiency.
- **Ecommerce and retail** Inventory and supply chain optimization, demand forecasting, visual search, personalized offers and experiences, and recommendation engines.
- **Financial services** Risk assessment and analysis, fraud detection, automated trading, and service processing optimization.
- **Telecommunications** Intelligent networks and network optimization, predictive maintenance, business process automation, upgrade planning, and capacity forecasting.

3 History of AI & ML

3.1 Precursors

Artificial intelligence is based on the assumption that the process of human thought can be mechanized. The study of mechanical - or "formal" - reasoning has a long history. Chinese, Indian and Greek philosophers all developed structured methods of formal deduction. Their ideas were developed over the centuries by philosophers such as Aristotle (who gave a formal analysis of the syllogism), Euclid (whose Elements was a model of formal reasoning), al-Khwārizmī (who developed algebra and gave his name to "algorithm").

Spanish philosopher **Ramon Llull** (1232–1315) developed several logical machines devoted to the production of knowledge by logical means; Llull described his machines as mechanical entities that could combine basic and undeniable truths by simple logical operations, produced by the machine by mechanical meanings, in such ways as to produce all the possible knowledge. Llull's work had a great influence on Gottfried Leibniz, who redeveloped his ideas.

In the 17th century, **Leibniz**, Thomas Hobbes and René Descartes explored the possibility that all rational thought could be made as systematic as algebra or geometry. Hobbes famously wrote in Leviathan: "reason is nothing but reckoning". Leibniz envisioned a universal language of reasoning, the characteristica universalis, which would reduce argumentation to calculation so that "there would be no more need of disputation between two philosophers than between two accountants. For it would suffice to take their pencils in hand, down to their slates, and to say each other (with a friend as witness, if they liked): Let us calculate". These philosophers had begun to articulate the physical symbol system hypothesis that would become the guiding faith of AI research.

In the 20th century, the study of mathematical logic provided the essential breakthrough that made artificial intelligence seem plausible. The foundations had been set by such works as Boole's The Laws of Thought and Frege's Begriffsschrift. Building on Frege's system, Russell and Whitehead presented a formal treatment of the foundations of mathematics in their masterpiece, the Principia Mathematica in 1913. Inspired by Russell's success, David Hilbert challenged mathematicians of the 1920s and 30s to answer this fundamental question: **"can all of mathematical reasoning be formalized?"**

His question was answered by Gödel's incompleteness proof, Turing's machine and Church's Lambda calculus.

Their answer was surprising in two ways. First, they proved that there were, in fact, limits to what mathematical logic could accomplish. But second (and more important for AI) their work suggested that, within these limits, any form of mathematical reasoning could be mechanized. The Church-Turing thesis implied that a mechanical device, shuffling symbols as simple as 0 and 1, could imitate any conceivable process of mathematical deduction. The key insight was the Turing machine - a simple theoretical construct that captured the essence of abstract symbol manipulation. This invention would inspire a handful of scientists to begin discussing the possibility of thinking machines.

3.2 The birth of AI

In the 1940s and 50s, a handful of scientists from a variety of fields (mathematics, psychology, engineering, economics and political science) began to discuss the possibility of creating an artificial brain. The field of artificial intelligence research was founded as an academic discipline in 1956.

Alan Turing In 1950 Alan Turing published a landmark paper in which he speculated about the possibility of creating machines that think. He noted that "thinking" is difficult to define and devised his famous Turing Test. If a machine could carry on a conversation (over a teleprinter) that was indistinguishable from a conversation with a human being, then it was reasonable to say that the machine was "thinking". This simplified version of the problem allowed Turing to argue convincingly that a "thinking machine" was at least plausible and the paper answered all the most common objections to the proposition. **The Turing Test was the first serious proposal in the philosophy of artificial intelligence**.

His paper was followed in 1952 by the Hodgkin-Huxley model of the brain as neurons forming an electrical network, with individual neurons firing in all-or-nothing (on/off) pulses. These combined events, discussed at a conference sponsored by Dartmouth College in 1956, helped to spark the concept of artificial intelligence.

Dartmouth Workshop The Dartmouth Workshop of 1956 was organized by Marvin Minsky, John McCarthy and two senior scientists: Claude Shannon and Nathan Rochester of IBM. The proposal for the conference included this assertion: "every aspect of learning or any other feature of intelligence can be so precisely described that a machine can be made to simulate it"[1]. The participants included Ray Solomonoff, Oliver Selfridge, Trenchard More, Arthur Samuel, Allen Newell and Herbert A. Simon, all of whom would create important programs during the first decades of AI research. **The 1956 Dartmouth workshop was the moment that AI gained its name**, its mission, its first success and its major players, and is widely considered the birth of AI. The term "Artificial Intelligence" was chosen by McCarthy to avoid associations with cybernetics and connections with the influential cyberneticist Norbert Wiener.

3.3 The First AI Winter (1974-1980)

The stretch of time between 1974 and 1980 has become known as 'The First AI Winter'. AI was subject to critiques and financial setbacks, AI researchers had failed to appreciate the difficulty of the problems they faced. Their tremendous optimism had raised expectations impossibly high, and when the promised results failed to materialize, funding for AI disappeared. At the same time, the field of connectionism (or neural nets) was shut down almost completely for 10 years by Marvin Minsky's devastating criticism of perceptrons.

The capabilities of AI programs were limited. Even the most impressive could only handle trivial versions of the problems they were supposed to solve; all the programs were, in some sense, "toys". AI researchers had begun to run into several fundamental limits that could not be overcome in the 1970s. Although some of these limits would be conquered in later decades, others still stymie the field to this day:

[1] http://jmc.stanford.edu/articles/dartmouth/dartmouth.pdf

- **Limited computer power**: There was not enough memory or processing speed to accomplish anything truly useful. For example, Ross Quillian's successful work on natural language was demonstrated with a vocabulary of only twenty words, because that was all that would fit in memory.
- **Intractability and the combinatorial explosion**: many problems that can probably only be solved in exponential time
- **Commonsense knowledge and reasoning**: Many important artificial intelligence applications like vision or natural language require simply enormous amounts of information about the world: the program needs to have some idea of what it might be looking at or what it is talking about. This requires that the program know most of the same things about the world that a child does. Researchers soon discovered that this was a truly vast amount of information. No one in 1970 could build a database so large and no one knew how a program might learn so much information.
- **Moravec's paradox**: Proving theorems and solving geometry problems is comparatively easy for computers, but a supposedly simple task like recognizing a face or crossing a room without bumping into anything is extremely difficult. This helps explain why research into vision and robotics had made so little progress by the middle 1970s.

Artificial intelligence research had its government funding cut, and interest dropped off. However, unlike gravity, AI research resumed in the 1980s, with the U.S. and Britain providing funding to compete with Japan's new "fifth generation" computer project, and their goal of becoming the world leader in computer technology.

The First AI Winter ended with the promising introduction of "Expert Systems," which were developed and quickly adopted by large competitive corporations all around the world. The primary focus of AI research was now on the theme of accumulating knowledge from various experts, and sharing that knowledge with its users. AI also benefited from the revival of Connectionism (or Neural nets) in the 1980s.

The term "AI winter" was coined by researchers who had survived the funding cuts of 1974 when they became concerned that enthusiasm for expert systems had spiraled out of control and that disappointment would certainly follow. Their fears were well founded: in the late 1980s and early 1990s, AI suffered a series of financial setbacks.

3.4 Rebirth of AI (1993-2011)

The field of AI, now more than a half a century old, finally achieved some of its oldest goals. It began to be used successfully throughout the technology industry, although somewhat behind the scenes. Some of the success was due to increasing computer power and some was achieved by focusing on specific isolated problems and pursuing them with the highest standards of scientific accountability. Still, the reputation of AI, in the business world at least, was less than pristine. Inside the field there was little agreement on the reasons for AI's failure to fulfill the dream of human level intelligence that had captured the imagination of the world in the 1960s. Together, all these factors helped to fragment AI into competing sub-fields focused on particular problems or approaches, sometimes even under new names that disguised the tarnished pedigree of "artificial intelligence". AI was both more cautious and more successful than it had ever been.

The Chess Game Game AI and in particular chess game was used as a measure of progress in AI throughout its history. On 11 May 1997, Deep Blue became the first computer chess-playing system to beat a reigning world chess champion, Garry Kasparov. The super computer was a specialized version of a framework produced by IBM, and was capable of processing twice as many moves per second as it had during the first match (which Deep Blue had lost), reportedly 200,000,000 moves per second. The event was broadcast live over the internet and received over 74 million hits.

DARPA Grand Challenge In 2005, a Stanford robot won the DARPA Grand Challenge by driving autonomously for 131 miles along an unrehearsed desert trail. Two years later, a team from CMU won the DARPA Urban Challenge by autonomously navigating 55 miles in an Urban environment while adhering to traffic hazards and all traffic laws.

Jeopardy In February 2011, in a Jeopardy! quiz show exhibition match, IBM's question answering system, Watson, defeated the two greatest Jeopardy! champions, Brad Rutter and Ken Jennings, by a significant margin.

These successes were not due to some revolutionary new paradigm, but mostly on the tedious application of engineering skill and on the tremendous increase in the speed and capacity of computer by the 90s. In fact, Deep Blue's computer was 10 million times faster than the Ferranti Mark 1 that Christopher Strachey taught to play chess in 1951. This dramatic increase is measured by Moore's law, which predicts that the speed and memory capacity of computers doubles every two years, as a result of metal–oxide–semiconductor (MOS) transistor counts doubling every two years. The fundamental problem of "raw computer power" was slowly being overcome.

3.5 The Artificial Intelligence nowadays

In the first decades of the 21st century, access to large amounts of data (known as "big data"), cheaper and faster computers and advanced machine learning techniques were successfully applied to many problems throughout the economy. In fact, McKinsey Global Institute estimated in their famous paper "Big data: The next frontier for innovation, competition, and productivity" that "by 2009, nearly all sectors in the US economy had at least an average of 200 terabytes of stored data". By 2016, the market for AI-related products, hardware, and software reached more than 8 billion dollars, and the New York Times reported that interest in AI had reached a "frenzy".
The applications of big data began to reach into other fields as well, such as training models in ecology and for various applications in economics. Advances in deep learning (particularly deep convolutional neural networks and recurrent neural networks) drove progress and research in image and video processing, text analysis, and even speech recognition.

Big Data Big data refers to a collection of data that cannot be captured, managed, and processed by conventional software tools within a certain time frame. It is a massive amount of decision-making, insight, and process optimization capabilities that require new processing models. In the Big Data Era written by Victor Meyer Schonberg and Kenneth Cooke, big data means that instead of random analysis (sample survey), all data is used for analysis. The 5V characteristics of big data (proposed by IBM):

- Volume
- Velocity
- Variety
- Value
- Veracity

The strategic significance of big data technology is not to master huge data information, but to specialize in these meaningful data. In other words, if big data is likened to an industry, the key to realizing profitability in this industry is to increase the "process capability" of the data and realize the "value added" of the data through "processing".

GAI General intelligence is the ability to solve any problem, rather than finding a solution to a particular problem. Artificial general intelligence (or "AGI") is a program which can apply intelligence to a wide variety of problems, in much the same ways humans can. Artificial general intelligence is also referred to as "strong AI", or synthetic intelligence as opposed to "weak AI" or "narrow AI". (Academic sources reserve "strong AI" to refer to machines capable of experiencing consciousness). Foundation models, which are large artificial intelligence models trained on vast quantities of unlabeled data that can be adapted to a wide range of downstream tasks, began to be developed in 2018. Models such as GPT-3 released by OpenAI in 2020, and Gato released by DeepMind in 2022, have been described as important milestones on the path to artificial general intelligence. In 2023, Microsoft Research tested the GPT-4 large language model with a large variety of tasks, and concluded that "it could reasonably be viewed as an early (yet still incomplete) version of an artificial general intelligence (AGI) system".

AI: Main Goals The general problem of simulating (or creating) intelligence has been broken down into sub-problems. These consist of particular traits or capabilities that researchers expect an intelligent system to display. The traits described below have received the most attention:

- **Reasoning, problem-solving**: Early researchers developed algorithms that imitated step-by-step reasoning that humans use when they solve puzzles or make logical deductions. By the late 1980s and 1990s, AI research had developed methods for dealing with uncertain or incomplete information, employing concepts from probability and economics. Many of these algorithms proved to be insufficient for solving large reasoning problems because they experienced a "combinatorial explosion": they became exponentially slower as the problems grew larger. Even humans rarely use the step-by-step deduction that early AI research could model. They solve most of their problems using fast, intuitive judgments.

- **Knowledge representation**: Knowledge representation and knowledge engineering allow AI programs to answer questions intelligently and make deductions about real-world facts. AI research has developed tools to represent specific domains, such as objects, properties, categories and relations between objects; situations, events, states and time; causes and effects; knowledge about knowledge (what we know about what other people know); default reasoning (things that humans assume are true until they are told differently and will remain true even when other facts are changing); as well as other domains. Among the most difficult problems in AI are: the breadth of commonsense knowledge (the number of atomic facts that the average person knows is enormous); and the sub-symbolic form of most commonsense knowledge (much of what people know is not represented as "facts" or "statements" that they could express verbally). Formal knowledge representations are used in content-based indexing and retrieval, scene interpretation, clinical decision support, knowledge discovery (mining "interesting" and actionable inferences from large databases) and other areas.

- **Machine Learning**: a fundamental concept of AI research since the field's inception, is the study of computer algorithms that improve automatically through experience.

- **Natural Language Processing**: allows machines to read and understand human language. A sufficiently powerful natural language processing system would enable natural-language user interfaces and the acquisition of knowledge directly from human-written sources, such as newswire texts. Some straightforward applications of NLP include information retrieval, question answering and machine translation. Symbolic AI used formal syntax to translate the deep structure of sentences into logic. This failed to produce useful applications, due to the intractability of logic and the breadth of commonsense knowledge. Modern statistical techniques include co-occurrence frequencies (how often one word appears near another), "Keyword spotting" (searching for a particular word to retrieve information), transformer-based deep learning (which finds patterns in text), and others. They have achieved acceptable accuracy at the page or paragraph level, and, by 2019, could generate coherent text.

- **Perception**: Machine perception is the ability to use input from sensors (such as cameras, microphones, wireless signals, and active lidar, sonar, radar, and tactile sensors) to deduce aspects of the world. Applications include speech recognition, facial recognition, and object recognition. Computer vision is the ability to analyze visual input.

- **General Intelligence**: A machine with general intelligence can solve a wide variety of problems with breadth and versatility similar to human intelligence. There are several competing ideas about how to develop artificial general intelligence.

- **Social Intelligence**: Affective computing is an interdisciplinary umbrella that comprises systems that recognize, interpret, process or simulate human feeling, emotion and mood. For example, some virtual assistants are programmed to speak conversationally or even to banter humorously; it makes them appear more sensitive to the emotional dynamics of human interaction, or to otherwise facilitate human–computer interaction. However, this tends to give naïve users an unrealistic conception of how intelligent existing computer agents actually are. Moderate successes related to affective computing include textual sentiment analysis and, more recently, multimodal sentiment analysis), wherein AI classifies the affects displayed by a videotaped subject.

4 Deeper in AI: Machine Learning

Machine learning (ML) is a field of inquiry devoted to understanding and building methods that "learn" - that is, methods that leverage data to improve performance on some set of tasks. It is seen as a part of artificial intelligence. Machine learning algorithms build a model based on sample data, known as training data, in order to make predictions or decisions without being explicitly programmed to do so. Machine learning algorithms are used in a wide variety of applications, such as in medicine, email filtering, speech recognition, agriculture, and computer vision, where it is difficult or unfeasible to develop conventional algorithms to perform the needed tasks.

A subset of machine learning is closely related to computational statistics, which focuses on making predictions using computers, but not all machine learning is statistical learning. The study of mathematical optimization delivers methods, theory and application domains to the field of machine learning. Data mining is a related field of study, focusing on exploratory data analysis through unsupervised learning.

Some implementations of machine learning use data and neural networks in a way that mimics the working of a biological brain.

In its application across business problems, machine learning is also referred to as predictive analytics.

4.1 Overview of ML

Learning algorithms work on the basis that strategies, algorithms, and inferences that worked well in the past are likely to continue working well in the future. These inferences can be obvious, such as "since the sun rose every morning for the last 10,000 days, it will probably rise tomorrow morning as well". They can be nuanced, such as "X% of families have geographically separate species with color variants, so there is a Y% chance that undiscovered black swans exist".

Machine learning programs can perform tasks without being explicitly programmed to do so. It involves computers learning from data provided so that they carry out certain tasks. For simple tasks assigned to computers, it is possible to program algorithms telling the machine how to execute all steps required to solve the problem at hand; on the computer's part, no learning is needed. For more advanced tasks, it can be challenging for a human to manually create the needed algorithms. In practice, it can turn out to be more effective to help the machine develop its own algorithm, rather than having human programmers specify every needed step.

The discipline of machine learning employs various approaches to teach computers to accomplish tasks where no fully satisfactory algorithm is available. In cases where vast numbers of potential answers exist, one approach is to label some of the correct answers as valid. This can then be used as training data for the computer to improve the algorithm(s) it uses to determine correct answers. For example, to train a system for the task of digital character recognition, the MNIST dataset of handwritten digits has often been used.

4.2 Approaches of ML

Machine learning approaches are traditionally divided into three broad categories, which correspond to learning paradigms, depending on the nature of the "signal" or "feedback" available to the learning system:

- Supervised learning: The computer is presented with example inputs and their desired outputs, given by a "teacher", and the goal is to learn a general rule that maps inputs to outputs.
- Unsupervised learning: No labels are given to the learning algorithm, leaving it on its own to find structure in its input. Unsupervised learning can be a goal in itself (discovering hidden patterns in data) or a means towards an end (feature learning).
- Reinforcement learning: A computer program interacts with a dynamic environment in which it must perform a certain goal (such as driving a vehicle or playing a game against an opponent). As it navigates its problem space, the program is provided feedback that's analogous to rewards, which it tries to maximize.

Supervised Learning Supervised learning algorithms build a mathematical model of a set of data that contains both the inputs and the desired outputs. The data is known as training data, and consists of a set of training examples. Each training example has one or more inputs and the desired output, also known as a supervisory signal. In the mathematical model, each training example is represented by an array or vector, sometimes called a feature vector, and the training data is represented by a matrix. Through iterative optimization of an objective function, supervised learning algorithms learn a function that can be used to predict the output associated with new inputs. An optimal function will allow the algorithm to correctly determine the output for inputs that were not a part of the training data. An algorithm that improves the accuracy of its outputs or predictions over time is said to have learned to perform that task. Types of supervised-learning algorithms include active learning, classification and regression. Classification algorithms are used when the outputs are restricted to a limited set of values, and regression algorithms are used when the outputs may have any numerical value within a range. As an example, for a classification algorithm that filters emails, the input would be an incoming email, and the output would be the name of the folder in which to file the email. Similarity learning is an area of supervised machine learning closely related to regression and classification, but the goal is to learn from examples using a similarity function that measures how similar or related two objects are. It has applications in ranking, recommendation systems, visual identity tracking, face verification, and speaker verification.

Unsupervised Learning Unsupervised learning algorithms take a set of data that contains only inputs, and find structure in the data, like grouping or clustering of data points. The algorithms, therefore, learn from test data that has not been labeled, classified or categorized. Instead of responding to feedback, unsupervised learning algorithms identify commonalities in the data and react based on the presence or absence of such commonalities in each new piece of data. A central application of unsupervised learning is in the field of density estimation in statistics, such as finding the probability density function. Though unsupervised learning encompasses other domains involving summarizing and explaining data features.

Cluster analysis is the assignment of a set of observations into subsets (called clusters) so that observations within the same cluster are similar according to one or more predesignated criteria, while observations drawn from different clusters are dissimilar. Different clustering techniques make different assumptions on the structure of the data, often defined by some similarity metric and evaluated, for example, by internal compactness, or the similarity between members of the same cluster, and separation, the difference between clusters. Other methods are based on estimated density and graph connectivity.

Semi-Supervised Learning Semi-supervised learning falls between unsupervised learning (without any labeled training data) and supervised learning (with completely labeled training data). Some of the training examples are missing training labels, yet many machine-learning researchers have found that unlabeled data, when used in conjunction with a small amount of labeled data, can produce a considerable improvement in learning accuracy.

In weakly supervised learning, the training labels are noisy, limited, or imprecise; however, these labels are often cheaper to obtain, resulting in larger effective training sets.

Reinforcement Learning Reinforcement learning is an area of machine learning concerned with how software agents ought to take actions in an environment so as to maximize some notion of cumulative reward. Due to its generality, the field is studied in many other disciplines, such as game theory, control theory, operations research, information theory, simulation-based optimization, multi-agent systems, swarm intelligence, statistics and genetic algorithms. In machine learning, the environment is typically represented as a Markov decision process (MDP). Many reinforcements learning algorithms use dynamic programming techniques. Reinforcement learning algorithms do not assume knowledge of an exact mathematical model of the MDP and are used when exact models are infeasible. Reinforcement learning algorithms are used in autonomous vehicles or in learning to play a game against a human opponent.

Dimensionality Reduction Dimensionality reduction is a process of reducing the number of random variables under consideration by obtaining a set of principal variables. In other words, it is a

process of reducing the dimension of the feature set, also called the "number of features". Most of the dimensionality reduction techniques can be considered as either feature elimination or extraction. One of the popular methods of dimensionality reduction is principal component analysis (PCA). PCA involves changing higher-dimensional data (e.g., 3D) to a smaller space (e.g., 2D). This results in a smaller dimension of data (2D instead of 3D), while keeping all original variables in the model without changing the data. The manifold hypothesis proposes that high-dimensional data sets lie along low-dimensional manifolds, and many dimensionality reduction techniques make this assumption, leading to the area of manifold learning and manifold regularization.

4.3 Key concepts of ML

Now that we have an understanding of the three main types of machine learning, let us discuss some key concepts for understanding how machine learning algorithms work. These include training and testing data, feature extraction, and model selection. It is important to understand these concepts as they are the building blocks of all machine learning algorithms.

Training data is the data that is used to train a machine learning algorithm. This data is used to build a model that can make predictions on new data. **Testing data** is a separate set of data that is used to evaluate the performance of the trained model. The goal is to build a model that performs well on both the training and testing data, which is a measure of the model's ability to generalize to new data.

Feature extraction is the process of transforming raw data into a set of features that can be used as inputs to a machine learning algorithm. Feature extraction is an important step in machine learning because it allows the algorithm to focus on the most relevant information in the data. For example, in image classification, the features could be the brightness, color, and texture of the image.

Model selection is the process of choosing the best algorithm and parameters for a particular problem. There are many different machine learning algorithms to choose from, each with its strengths and weaknesses. Additionally, each algorithm has parameters that can be tuned to optimize its performance for a particular problem.

Model Assessments is the process to validate machine learning models by accuracy estimation techniques like the holdout method, which splits the data in a training and test set (conventionally 2/3 training set and 1/3 test set designation) and evaluates the performance of the training model on the test set. In comparison, the K-fold-cross-validation method randomly partitions the data into K subsets and then K experiments are performed each respectively considering 1 subset for evaluation and the remaining K-1 subsets for training the model. In addition to the holdout and cross-validation methods, bootstrap, which samples n instances with replacement from the dataset, can be used to assess model accuracy. In addition to overall accuracy, investigators frequently report sensitivity and specificity meaning True Positive Rate (TPR) and True Negative Rate (TNR) respectively. Similarly, investigators sometimes report the false positive rate (FPR) as well as the false negative rate (FNR). However, these rates are ratios that fail to reveal their numerators and denominators. The total operating characteristic (TOC) is an effective method to express a model's diagnostic ability. TOC shows the numerators and denominators of the previously mentioned rates, thus TOC provides more information than the commonly used receiver operating characteristic (ROC) and ROC's associated area under the curve (AUC).

While machine learning has many benefits, there are also potential pitfalls.

4.4 Limitations of ML

Although machine learning has been transformative in some fields, machine-learning programs often fail to deliver expected results. Reasons for this are numerous: lack of (suitable) data, lack of access to the data, data bias, privacy problems, badly chosen tasks and algorithms, wrong tools and people, lack of resources, and evaluation problems.

In 2018, a self-driving car from Uber failed to detect a pedestrian, who was killed after a collision. Attempts to use machine learning in healthcare with the IBM Watson system failed to deliver even after years of time and billions of dollars invested.

Machine learning has been used as a strategy to update the evidence related to a systematic review and increased reviewer burden related to the growth of biomedical literature. While it has improved with training sets, it has not yet developed sufficiently to reduce the workload burden without limiting the necessary sensitivity for the findings research themselves.

Bias Machine learning approaches in particular can suffer from different data biases. A machine learning system trained specifically on current customers may not be able to predict the needs of new customer groups that are not represented in the training data. When trained on human-made data, machine learning is likely to pick up the constitutional and unconscious biases already present in society. Language models learned from data have been shown to contain human-like biases. Machine learning systems used for criminal risk assessment have been found to be biased against black people. In 2015, Google photos would often tag black people as gorillas, and in 2018 this still was not well resolved, but Google reportedly was still using the workaround to remove all gorillas from the training data, and thus was not able to recognize real gorillas at all. Similar issues with recognizing non-white people have been found in many other systems. In 2016, Microsoft tested a chatbot that learned from Twitter, and it quickly picked up racist and sexist language. Because of such challenges, the effective use of machine learning may take longer to be adopted in other domains. Concern for fairness in machine learning, that is, reducing bias in machine learning and propelling its use for human good is increasingly expressed by artificial intelligence scientists, including Fei-Fei Li, who reminds engineers that "There's nothing artificial about AI...It's inspired by people, it's created by people, and—most importantly—it impacts people. It is a powerful tool we are only just beginning to understand, and that is a profound responsibility".

Explainability Explainable AI (XAI), or Interpretable AI, or Explainable Machine Learning (XML), is artificial intelligence (AI) in which humans can understand the decisions or predictions made by the AI. It contrasts with the "black box" concept in machine learning where even its designers cannot explain why an AI arrived at a specific decision. By refining the mental models of users of AI-powered systems and dismantling their misconceptions, XAI promises to help users perform more effectively. XAI may be an implementation of the social right to explanation.

Overfitting One of the most significant pitfalls is overfitting, where a model is too complex and fits the training data too closely, resulting in poor performance on new data. Underfitting is the opposite problem, where a model is too simple and fails to capture the underlying patterns in the data. It is important to carefully balance the complexity of the model with the amount of data available to avoid these issues. Many systems attempt to reduce overfitting by rewarding a theory in accordance with how well it fits the data but penalizing the theory in accordance with how complex the theory is.

4.5 Models of ML

Performing machine learning involves creating a model, which is trained on some training data and then can process additional data to make predictions. Various types of models have been used and researched for machine learning systems:

- Artificial neural networks (ANN)
- Decision trees
- Support-vector machines (SVM)
- Regression analysis
- Bayesian networks
- Gaussian processes
- Genetic algorithms (GA)

- and more...

Let's see in detail some of these most important algorithms of Machine Learning.

Regression Analysis In statistical modeling, regression analysis is a set of statistical processes for estimating the relationships between a dependent variable (often called the 'outcome' or 'response' variable, or a 'label' in machine learning parlance) and one or more independent variables (often called 'predictors', 'covariates', 'explanatory variables' or 'features'). The most common form of regression analysis is linear regression, in which one finds the line (or a more complex linear combination) that most closely fits the data according to a specific mathematical criterion. For example, the method of ordinary least squares computes the unique line (or hyperplane) that minimizes the sum of squared differences between the true data and that line (or hyperplane). For specific mathematical reasons, this allows the researcher to estimate the conditional expectation (or population average value) of the dependent variable when the independent variables take on a given set of values. Less common forms of regression use slightly different procedures to estimate alternative location parameters (e.g., quantile regression or Necessary Condition Analysis) or estimate the conditional expectation across a broader collection of non-linear models (e.g., nonparametric regression).

Regression analysis is primarily used for two conceptually distinct purposes:

1. First, regression analysis is widely used for prediction and forecasting, where its use has substantial overlap with the field of machine learning.

2. Second, in some situations regression analysis can be used to infer causal relationships between the independent and dependent variables.

Importantly, regressions by themselves only reveal relationships between a dependent variable and a collection of independent variables in a fixed dataset. To use regressions for prediction or to infer causal relationships, respectively, a researcher must carefully justify why existing relationships have predictive power for a new context or why a relationship between two variables has a causal interpretation. The latter is especially important when researchers hope to estimate causal relationships using observational data.

In practice, researchers first select a model they would like to estimate and then use their chosen method (e.g., ordinary least squares) to estimate the parameters of that model. Regression models involve the following components:

- The unknown parameters, often denoted as a scalar or vector β
- The independent variables, which are observed in data and are often denoted as a vector X_i (where i denotes a row of data)
- The dependent variable, which are observed in data and often denoted using the scalar Y_i
- The error terms, which are not directly observed in data and are often denoted using the scalar e_i

In various fields of application, different terminologies are used in place of dependent and independent variables. Most regression models propose that Y_i is a function of X_i and β, with e_i representing an additive error term that may stand in for un-modeled determinants of Y_i or random statistical noise:

$$Y_i = f(X_i, \beta) + e_i$$

The researchers' goal is to estimate the function $f(X_i, \beta)$ that most closely fits the data. To carry out regression analysis, the form of the function f must be specified. Sometimes the form of this function is based on knowledge about the relationship between Y_i and X_i that does not rely on the data. If no such knowledge is available, a flexible or convenient form for f is chosen. For example, a simple univariate regression may propose $f(X_i, \beta) = \beta_0 + \beta_1 X_i$, suggesting that the researcher believes $Y_i = \beta_0 + \beta_1 X_i + e_i$ to be a reasonable approximation for the statistical process generating the data. Once researchers determine their preferred statistical model, different forms of regression

analysis provide tools to estimate the parameters β. For example, least squares (including its most common variant, ordinary least squares) finds the value of β that minimizes the sum of squared errors:

$$\sum_i (Y_i - f(X_i, \beta))^2$$

A given regression method will ultimately provide an estimate of β, usually denoted $\hat{\beta}$ to distinguish the estimate from the true (unknown) parameter value that generated the data. Using this estimate, the researcher can then use the fitted value $\hat{Y}_i = f(X_i, \hat{\beta})$ for prediction or to assess the accuracy of the model in explaining the data. Whether the researcher is intrinsically interested in the estimate $\hat{\beta}$ or the predicted value $\hat{Y}_i$ will depend on context and their goals. As described in ordinary least squares, least squares is widely used because the estimated function $f(X_i, \hat{\beta})$ approximates the conditional expectation $E(Y_i|X_i)$.
The most common and used regression is the Linear regression, where the dependent variable y_i is a linear combiation of the **parameters**[2].

By itself, a regression is simply a calculation using the data. In order to interpret the output of regression as a meaningful statistical quantity that measures real-world relationships, researchers often rely on a number of classical assumptions. These assumptions often include:

- The sample is representative of the population at large
- The independent variables are measured with no error
- Deviations from the model have an expected value of zero, conditional on covariates: $E(e_i|X_i) = 0$
- The variance of the residuals e_i is constant across observations (homoscedasticity)
- The residuals e_i are uncorrelated with one another. Mathematically, the variance–covariance matrix of the errors is diagonal.

In conclusion, Regression analysis encompasses a large variety of statistical methods to estimate the relationship between input variables and their associated features. Its most common form is linear regression, where a single line is drawn to best fit the given data according to a mathematical criterion such as ordinary least squares. The latter is often extended by regularization methods to mitigate overfitting and bias, as in ridge regression. When dealing with non-linear problems, go-to models include polynomial regression (for example, used for trendline fitting in Microsoft Excel), logistic regression (often used in statistical classification) or even kernel regression, which introduces non-linearity by taking advantage of the kernel trick to implicitly map input variables to higher-dimensional space.

Decision Trees Decision tree learning is a supervised learning approach used in statistics, data mining and machine learning. In this formalism, a classification or regression decision tree is used as a predictive model to draw conclusions about a set of observations. Tree models where the target variable can take a discrete set of values are called classification trees; in these tree structures, **leaves** represent class labels and **branches** represent conjunctions of features that lead to those class labels. Decision trees where the target variable can take continuous values (typically real numbers) are called regression trees. More generally, the concept of regression tree can be extended to any kind of object equipped with pairwise dissimilarities such as categorical sequences. Decision trees are among the most popular machine learning algorithms given their intelligibility and simplicity. In decision analysis, a decision tree can be used to visually and explicitly represent decisions and decision making. In data mining, a decision tree describes data (but the resulting classification tree can be an input for decision making). Decision tree learning is a method commonly used in data mining. The goal is to create a model that predicts the value of a target variable based on several input variables.

A decision tree is a simple representation for classifying examples. For this section, assume that all of the input features have finite discrete domains, and there is a single target feature called the "classification". Each element of the domain of the classification is called a class. A decision tree or a

[2]Note that adding a quadratic independent variable x_i is still linear regression

classification tree is a tree in which each internal (non-leaf) node is labeled with an input feature. The arcs coming from a node labeled with an input feature are labeled with each of the possible values of the target feature or the arc leads to a subordinate decision node on a different input feature. Each leaf of the tree is labeled with a class or a probability distribution over the classes, signifying that the data set has been classified by the tree into either a specific class, or into a particular probability distribution (which, if the decision tree is well-constructed, is skewed towards certain subsets of classes).

A tree is built by splitting the source set, constituting the root node of the tree, into subsets—which constitute the successor children. The splitting is based on a set of splitting rules based on classification features. This process is repeated on each derived subset in a recursive manner called **recursive partitioning**. The recursion is completed when the subset at a node has all the same values of the target variable, or when splitting no longer adds value to the predictions. This process of top-down induction of decision trees (TDIDT) is an example of a greedy algorithm, and it is by far the most common strategy for learning decision trees from data.

In data mining, decision trees can be described also as the combination of mathematical and computational techniques to aid the description, categorization and generalization of a given set of data.

Data comes in records of the form:

$$(x, Y) = (x_1, x_2, ..., x_k, Y)$$

The dependent variable Y is the target variable that we are trying to understand, classify or generalize. The vector x is composed of the features $x_1, x_2, ..., x_k$ that are used for that task.

Decision trees used in data mining are of two main types:

- **Classification tree** analysis is when the predicted outcome is the class (discrete) to which the data belongs.
- **Regression tree** analysis is when the predicted outcome can be considered a real number (e.g. the price of a house, or a patient's length of stay in a hospital).

The term classification and regression tree (CART) analysis is an umbrella term used to refer to either of the above procedures. Trees used for regression and trees used for classification have some similarities – but also some differences, such as the procedure used to determine where to split. Some techniques, often called ensemble methods, construct more than one decision tree:

- **Boosted trees** Incrementally building an ensemble by training each new instance to emphasize the training instances previously mis-modeled. A typical example is AdaBoost. These can be used for regression-type and classification-type problems.
- **Bootstrap aggregated** (or bagged) decision trees, an early ensemble method, builds multiple decision trees by repeatedly resampling training data with replacement, and voting the trees for a consensus prediction. A **random forest** classifier is a specific type of bootstrap aggregating
- **Rotation forest** in which every decision tree is trained by first applying principal component analysis (PCA) on a random subset of the input features.

Algorithms for constructing decision trees usually work top-down, by choosing a variable at each step that best splits the set of items. Different algorithms use different metrics for measuring "best". These generally measure the homogeneity of the target variable within the subsets. These metrics are applied to each candidate subset, and the resulting values are combined (e.g., averaged) to provide a measure of the quality of the split. Depending on the underlying metric, the performance of various heuristic algorithms for decision tree learning may vary significantly.

Advantages

Amongst other data mining methods, decision trees have various advantages:

- Simple to understand and interpret. People are able to understand decision tree models after a brief explanation. Trees can also be displayed graphically in a way that is easy for non-experts to interpret.

- Able to handle both numerical and categorical data. Other techniques are usually specialized in analyzing datasets that have only one type of variable. (For example, relation rules can be used only with nominal variables while neural networks can be used only with numerical variables or categoricals converted to 0-1 values.) Early decision trees were only capable of handling categorical variables, but more recent versions, such as C4.5, do not have this limitation.
- Requires little data preparation. Other techniques often require data normalization. Since trees can handle qualitative predictors, there is no need to create dummy variables.
- Uses a white box or open-box model. If a given situation is observable in a model the explanation for the condition is easily explained by boolean logic. By contrast, in a black box model, the explanation for the results is typically difficult to understand, for example with an artificial neural network.
- Possible to validate a model using statistical tests. That makes it possible to account for the reliability of the model.
- Non-parametric approach that makes no assumptions of the training data or prediction residuals; e.g., no distributional, independence, or constant variance assumptions
- Performs well with large datasets. Large amounts of data can be analyzed using standard computing resources in reasonable time.
- Accuracy with flexible modeling. These methods may be applied to healthcare research with increased accuracy.
- Mirrors human decision making more closely than other approaches. This could be useful when modeling human decisions/behavior.
- Robust against co-linearity, particularly boosting.
- In built feature selection. Additional irrelevant feature will be less used so that they can be removed on subsequent runs. The hierarchy of attributes in a decision tree reflects the importance of attributes. It means that the features on top are the most informative.
- Decision trees can approximate any Boolean function e.g. XOR.

Limitations

- Trees can be very non-robust. A small change in the training data can result in a large change in the tree and consequently the final predictions.
- The problem of learning an optimal decision tree is known to be NP-complete under several aspects of optimality and even for simple concepts. Consequently, practical decision-tree learning algorithms are based on heuristics such as the greedy algorithm where locally optimal decisions are made at each node. Such algorithms cannot guarantee to return the globally optimal decision tree. To reduce the greedy effect of local optimality, some methods such as the dual information distance (DID) tree were proposed.
- Decision-tree learners can create over-complex trees that do not generalize well from the training data. (This is known as overfitting). Mechanisms such as pruning are necessary to avoid this problem (with the exception of some algorithms such as the Conditional Inference approach, that does not require pruning).
- The average depth of the tree that is defined by the number of nodes or tests till classification is not guaranteed to be minimal or small under various splitting criteria.
- For data including categorical variables with different numbers of levels, information gain in decision trees is biased in favor of attributes with more levels. To counter this problem, instead of choosing the attribute with highest information gain, one can choose the attribute with the highest information gain ratio among the attributes whose information gain is greater than the mean information gain. This biases the decision tree against considering attributes with a

large number of distinct values, while not giving an unfair advantage to attributes with very low information gain. Alternatively, the issue of biased predictor selection can be avoided by the Conditional Inference approach, a two-stage approach, or adaptive leave-one-out feature selection.

Artificial Neural Network Artificial neural networks (ANNs), or connectionist systems, are computing systems vaguely inspired by the biological neural networks that constitute animal brains. Such systems "learn" to perform tasks by considering examples, generally without being programmed with any task-specific rules. More details later on...

5 Deep Learning: Artificial Neural Network

Neural networks and deep learning are two topics that have been gaining a lot of attention in recent years due to their potential to transform various industries, including healthcare, finance, transportation, and more. In this chapter, we will explore the foundations of neural networks and deep learning, their types, architectures, and practical examples of how they are used in the real world.

5.1 What is Deep Learning?

Deep learning is a class of machine learning algorithms that uses multiple layers to progressively extract higher-level features from the raw input. For example, in image processing, lower layers may identify edges, while higher layers may identify the concepts relevant to a human such as digits or letters or faces.

From another angle to view deep learning, deep learning refers to 'computer-simulate' or 'automate' human learning processes from a source (e.g., an image of dogs) to a learned object (dogs). Therefore, a notion coined as "deeper" learning or "deepest" learning makes sense. The deepest learning refers to the fully automatic learning from a source to a final learned object. A deeper learning thus refers to a mixed learning process: a human learning process from a source to a learned semi-object, followed by a computer learning process from the human learned semi-object to a final learned object.
Most modern deep learning models are based on artificial neural networks, specifically convolutional neural networks (CNN)s, although they can also include propositional formulas or latent variables organized layer-wise in deep generative models such as the nodes in deep belief networks and deep Boltzmann machines.

In deep learning, each level learns to transform its input data into a slightly more abstract and composite representation. In an image recognition application, the raw input may be a matrix of pixels; the first representational layer may abstract the pixels and encode edges; the second layer may compose and encode arrangements of edges; the third layer may encode a nose and eyes; and the fourth layer may recognize that the image contains a face. Importantly, a deep learning process can learn which features to optimally place in which level on its own. This does not eliminate the need for hand-tuning; for example, varying numbers of layers and layer sizes can provide different degrees of abstraction.

The word "deep" in "deep learning" refers to the number of layers through which the data is transformed. More precisely, deep learning systems have a substantial credit assignment path (CAP) depth. The CAP is the chain of transformations from input to output. CAPs describe potentially causal connections between input and output. For a feedforward neural network, the depth of the CAPs is that of the network and is the number of hidden layers plus one (as the output layer is also parameterized). For recurrent neural networks, in which a signal may propagate through a layer more than once, the CAP depth is potentially unlimited. No universally agreed-upon threshold of depth divides shallow learning from deep learning, but most researchers agree that deep learning involves CAP depth higher than 2. CAP of depth 2 has been shown to be a universal approximator in the sense that it can emulate any function. Beyond that, more layers do not add to the function approximator ability of the network. Deep models (CAP ¿ 2) are able to extract better features than shallow models and hence, extra layers help in learning the features effectively.

Deep learning architectures can be constructed with a greedy layer-by-layer method. Deep learning helps to disentangle these abstractions and pick out which features improve performance.

For supervised learning tasks, deep learning methods eliminate feature engineering, by translating the data into compact intermediate representations akin to principal components, and derive layered structures that remove redundancy in representation.

Deep learning algorithms can be applied to unsupervised learning tasks. This is an important benefit because unlabeled data are more abundant than the labeled data. Examples of deep structures that can be trained in an unsupervised manner are deep belief networks.

5.2 Artificial Neural Network

Artificial neural networks (ANNs), usually simply called neural networks (NNs) or neural nets, are computing systems inspired by the biological neural networks that constitute animal brains.

An ANN is based on a collection of connected units or nodes called artificial neurons, which loosely model the neurons in a biological brain. Each connection, like the synapses in a biological brain, can transmit a signal to other neurons. An artificial neuron receives signals then processes them and can signal neurons connected to it. The "signal" at a connection is a real number, and the output of each neuron is computed by some non-linear function of the sum of its inputs. The connections are called edges. Neurons and edges typically have a weight that adjusts as learning proceeds. The weight increases or decreases the strength of the signal at a connection. Neurons may have a threshold such that a signal is sent only if the aggregate signal crosses that threshold.

Typically, neurons are aggregated into layers. Different layers may perform different transformations on their inputs. Signals travel from the first layer (the input layer), to the last layer (the output layer), possibly after traversing the layers multiple times.

Training Neural networks learn (or are trained) by processing examples, each of which contains a known "input" and "result," forming probability-weighted associations between the two, which are stored within the data structure of the net itself. The training of a neural network from a given example is usually conducted by determining the difference between the processed output of the network (often a prediction) and a target output. This difference is the error. The network then adjusts its weighted associations according to a learning rule and using this error value. Successive adjustments will cause the neural network to produce output that is increasingly similar to the target output. After a sufficient number of these adjustments, the training can be terminated based on certain criteria. This is a form of supervised learning.

Such systems "learn" to perform tasks by considering examples, generally without being programmed with task-specific rules. For example, in image recognition, they might learn to identify images that contain cats by analyzing example images that have been manually labeled as "cat" or "no cat" and using the results to identify cats in other images. They do this without any prior knowledge of cats, for example, that they have fur, tails, whiskers, and cat-like faces. Instead, they automatically generate identifying characteristics from the examples that they process.

ANNs began as an attempt to exploit the architecture of the human brain to perform tasks that conventional algorithms had little success with. They soon reoriented towards improving empirical results, abandoning attempts to remain true to their biological precursors. **ANNs have the ability to learn and model non-linearities and complex relationships**. This is achieved by neurons being connected in various patterns, allowing the output of some neurons to become the input of others. The network forms a directed, weighted graph.

An artificial neural network consists of simulated neurons. Each neuron is connected to other nodes via links like a biological axon-synapse-dendrite connection. All the nodes connected by links take in some data and use it to perform specific operations and tasks on the data. Each link has a weight, determining the strength of one node's influence on another, allowing weights to choose the signal between neurons.

Artificial Neurons ANNs are composed of artificial neurons which are conceptually derived from biological neurons. Each artificial neuron has inputs and produces a single output which can be sent to multiple other neurons. The inputs can be the feature values of a sample of external data, such as images or documents, or they can be the outputs of other neurons. The outputs of the final output neurons of the neural net accomplish the task, such as recognizing an object in an image.

To find the output of the neuron we take the weighted sum of all the inputs, weighted by the weights of the connections from the inputs to the neuron. We add a bias term to this sum. This weighted sum is sometimes called the **activation**. This weighted sum is then passed through a (usually nonlinear) activation function to produce the output. The initial inputs are external data, such as images and documents. The ultimate outputs accomplish the task, such as recognizing an object in an image. The neurons are typically organized into multiple layers, especially in deep learning. Neurons of one layer connect only to neurons of the immediately preceding and immediately following layers. The layer that receives external data is the input layer. The layer that produces the ultimate result is the output layer. In between them are zero or more hidden layers. Single layer and unlayered networks are also used. Between two layers, multiple connection patterns are possible. They can be 'fully connected', with every neuron in one layer connecting to every neuron in the next layer. They can be pooling, where a group of neurons in one layer connects to a single neuron in the next layer, thereby reducing the number of neurons in that layer. Neurons with only such connections form a directed acyclic graph and are known as **feedforward networks**. Alternatively, networks that allow connections between neurons in the same or previous layers are known as **recurrent networks**.

Hyperparameters A hyperparameter is a constant parameter whose value is set **before** the learning process begins. The values of parameters are derived via learning. Examples of hyperparameters include:

- **Number of hidden layers**: Hidden layers are the layers between input layer and output layer. *"Very simple. Just keep adding layers until the test error does not improve anymore"*. Many hidden units within a layer with regularization techniques can increase accuracy. Smaller number of units may cause underfitting.
- **Dropout**: is regularization technique to avoid overfitting (increase the validation accuracy) thus increasing the generalizing power. Generally, use a small dropout value of 20%-50% of neurons with 20% providing a good starting point. A probability too low has minimal effect and a value too high results in under-learning by the network. You are likely to get better performance when dropout is used on a larger network, giving the model more of an opportunity to learn independent representations.
- **Network Weight Initialization**: Ideally, it may be better to use different weight initialization schemes according to the activation function used on each layer. Mostly uniform distribution is used.
- **Activation function**: Activation functions are used to introduce nonlinearity to models, which allows deep learning models to learn nonlinear prediction boundaries. Generally, the rectifier activation function is the most popular. *Sigmoid* is used in the output layer while making binary predictions. *Softmax* is used in the output layer while making multi-class predictions.
- **Number of epochs**: is the number of times the whole training data is shown to the network while training. Increase the number of epochs until the validation accuracy starts decreasing even when training accuracy is increasing(overfitting).
- **Learning rate**: The learning rate defines the size of the corrective steps that the model takes to adjust for errors in each observation. A high learning rate shortens the training time, but with lower ultimate accuracy, while a lower learning rate takes longer, but with the potential for greater accuracy. Optimizations such as Quickprop are primarily aimed at speeding up error minimization, while other improvements mainly try to increase reliability. In order to avoid oscillation inside the network such as alternating connection weights, and to improve the rate of convergence, refinements use an adaptive learning rate that increases or decreases as appropriate. The concept of momentum allows the balance between the gradient and the previous change to be weighted such that the weight adjustment depends to some degree on the previous change. A momentum close to 0 emphasizes the gradient, while a value close to 1 emphasizes the last change.
- **Batch size**: Mini batch size is the number of sub samples given to the network after which parameter update happens. A good default for batch size might be 32. Also try 32, 64, 128, 256, and so on.

The values of some hyperparameters can be dependent on those of other hyperparameters. For example, the size of some layers can depend on the overall number of layers. There are some methods used to find out Hyperparameters, but sometimes they require significant computing and memory resources, for example: Manual Search, Grid Search, Random Search, Bayesian Optimization.

Learning process Learning is the adaptation of the network to better handle a task by considering sample observations. Learning involves adjusting the weights (and optional thresholds) of the network to improve the accuracy of the result. This is done by minimizing the observed errors. Learning is complete when examining additional observations does not usefully reduce the error rate. Even after learning, the error rate typically does not reach 0. If after learning, the error rate is too high, the network typically must be redesigned. Practically this is done by defining a cost function that is evaluated periodically during learning. As long as its output continues to decline, learning continues. The cost is frequently defined as a statistic whose value can only be approximated. The outputs are actually numbers, so when the error is low, the difference between the output (almost certainly a cat) and the correct answer (cat) is small. Learning attempts to reduce the total of the differences across the observations. Most learning models can be viewed as a straightforward application of optimization theory and statistical estimation.

Types of NN Neural networks can be classified into three main types:

1. **Feedforward neural networks** are the most basic type of neural network, and they are used for tasks such as image classification and language processing. In a feedforward neural network, information flows in only one direction, from the input layer to the output layer. Each layer of the network processes the input data and passes it to the next layer until the output layer is reached.

2. **Recurrent neural networks** are used for tasks such as speech recognition and language translation. In a recurrent neural network, information can flow in both directions, and the network has the ability to remember previous inputs. This makes recurrent neural networks particularly useful for processing sequences of data, such as sentences or time series data.

3. **Convolutional neural networks** are used for image and video processing tasks. In a convolutional neural network, the input data is processed by a series of convolutional layers, which extract features from the image or video. The output of these layers is then passed to a fully connected layer, which performs the final classification.

In practice Now, let us move on to practical examples of how machine learning is used in the real world. There are numerous applications of machine learning, ranging from healthcare to finance to self-driving cars.

One of the most significant applications of machine learning is in **healthcare**. Machine learning algorithms can be used to predict disease outcomes, identify high-risk patients, and develop personalized treatment plans. For example, machine learning has been used to predict patient readmissions, identify patients at risk of sepsis, and diagnose skin cancer.

In **finance**, machine learning algorithms are used to predict market trends, identify fraudulent transactions, and make investment decisions. Machine learning can also be used for credit risk analysis, customer segmentation, and fraud detection. For example, machine learning has been used to identify credit card fraud and predict stock prices.

In **transportation**, machine learning is being used to develop self-driving cars. Self-driving cars rely on machine learning algorithms to process sensor data and make driving decisions. Machine learning can also be used for traffic prediction, route optimization, and demand forecasting. For example, machine learning has been used to predict traffic patterns and optimize public transportation routes.

In **manufacturing**, machine learning is used for predictive maintenance, quality control, and supply chain management. Machine learning can also be used to optimize production processes, reduce

waste, and improve efficiency. For example, machine learning has been used to predict equipment failures and optimize production schedules.

6 Conclusion

In conclusion, machine learning is a powerful tool that has the potential to revolutionize the way we live and work. With the ability to learn from data, machine learning algorithms can make predictions and decisions that were previously impossible. However, it is important to carefully consider the potential pitfalls and limitations of machine learning, and to ensure that algorithms are developed and used responsibly.

One of the key challenges associated with the rapid development of AI and ML is the potential **loss of jobs** due to automation. As machines become more capable of performing tasks that were previously done by humans, there is a risk that many jobs will become obsolete, leading to unemployment and social inequality.

Another challenge is the **ethical implications** of using AI for decision-making: can raise ethical concerns, especially when it comes to issues such as privacy, bias, and fairness. For example, if an AI system is used to make decisions about employment, credit, or healthcare, there is a risk that the system may discriminate against certain groups of people, based on factors such as race, gender, or age. This highlights the importance of developing AI systems that are transparent, accountable, and free from bias.

Another challenge in AI and ML is the need for more diverse and **inclusive representation** in the field. Historically, the field of computer science has been dominated by white men, which has led to a lack of diversity and a lack of consideration for the social and ethical implications of AI. Increasing diversity in the field can help to ensure that AI and ML are developed in ways that are beneficial for everyone, rather than just a small group of people.

7 DID WE PASS THE ALAN TURING'S TEST?