Machine Learning

Lesson 1

Introduction to Machine Learning

Lecturer: Dr. Msagha J Mbogholi, PhD

Content

- Introduction to Machine Learning (ML)
- Definitions
- ML Design
- Example Applications
- Categories of ML

Part 1

Introduction

Introduction

- Michael Jordan is arguably a household name; whether you a basketball fan or not, you most likely have heard of this name.
- He is arguably the greatest basketball player of all time. However, the world did not really know him until he won his first NBA title with the Chicago Bulls in 1991; he went on to win five more titles with the same team before his retirement as a player.
- Before winning his first title, the Eastern Conference had been dominated by the Detroit "Bad Boys" Pistons led by Isaiah Thomas. This was the only team that was able to contain the Bulls for 3 years in a row; how did they do this?
- In order to understand how hard this was, understand that Jordan was literally a scoring machine, leading the league in scoring for 10 seasons when he was a player.
- The Pistons learnt that they had to contain him in every game, coming up with the infamous "Jordan rules"; these were effective those 3 seasons they beat the Bulls in the Eastern Conference Finals.
- However, in 1991 the Bulls won the Eastern Conference Finals against the Pistons, since it was Jordan's turn to learn how to outmaneuver the Jordan rules. One of the Pistons players pointed out that Jordan learnt that he had to involve his teammates in order to beat the Jordan rules, and this worked effectively; the Bulls swept the Pistons (4-0) in that Finals.

Introduction

- Notice how many times the word “learned” has been used in the previous slide?
- Human beings learn to do tasks better based on experiences, i.e., they gain knowledge based on their experiences.
- The Detroit Pistons coach Chuck Daly learned from losing to the Bulls more than once that they only needed to lock out Jordan from the game and they would effectively win it…hence the Jordan rules.
- Jordan learnt that he could not win a title by being the only scorer in the game and started learning to involve his teammates more, thus even when he had low points the rest of the team contributed enough points for the Bulls to win the game.
- How many times have you found yourself consciously not doing something because of a bad experience with it? You know the saying “been there, done that” right?
- So, can we transfer this experience “model” into the computing world, and get a computer to also learn from experiences in order to improve its efficiency or performance in handling tasks?

Introduction (cont'd)

- If you believe it can be done then you're in the right place; this is the essence of machine learning.
- There is learning in every aspect of life, and the most practical learning is done through experiential.
- It is true that we also learn a lot from the experiences of others, and in school we were taught (is that the same as learning?) by our teachers/instructors.
- You are now learning more about machine learning (ML). Coincidentally you may find the following resource very informative on the differences between teaching and learning: http://www.differencebetween.net/language/words-language/difference-between-teaching-and-learning/

Why ML?

- A key question that may be running through your mind is, why ML?
- You can easily write an algorithm and subsequently the code in a given language say, Python?
- Yes this is true, and is done a lot in the computing domain.
- However, there are a number of reasons that this approach will not be sufficient in developing a solution to certain problems.

Why ML? (cont'd)

- Have you ever thought about the process of image identification by human beings? You see your friend Ted walking towards you from a distance and you can tell that is him; how did you know? There are possibly hundreds of people walking towards you yet you picked him out from the many…how did you do that?
- Consider the situation when you travel to a foreign country; they speak with a different accent and intonations. However, you are able to understand what they are saying…have you thought about how many different adjustments your brain had to make for this to happen?
- Can you write an algorithm and subsequent code in a language like Java or Python to solve the problems above? Of course not.
- The reason for this is that an algorithm is written step by step action to solve the problem; clearly to write such an algorithm is an uphill task, to say the least.
- ML helps to solve such problems that involve what human beings and animals do routinely without much thought, that is to say, routinely.

Why ML? (cont'd)

- In this digital age there is a lot of data out there that can be put to better use by ML techniques.
- Consider the information that is extracted by search engines…gold mine there. How about information that is stored in large datasets which is the big data buzz of today, for example medical data that helps to determine patterns that diseases take?
- There is so much information in the digisphere that provides opportunities for ML.
- Can a simple program give meaning to this data? No. There is need for ML techniques to extract the data and give meaning to it.
- This is the second reason for using machine learning; to be able to extract data from large datasets, give meaning to it, and use it to solve problems.

Part 2

Definitions

Definitions

- Now that we have an idea of what machine learning is all about let us introduce the terms associated with it, starting with its own definition.
- As can be seen machine learning is about machines "learning", hence the emphasis and importance is defining what is meant by learning in this context.
- Mitchell (1997) puts it succinctly by stating that:
- "A computer program is said to learn from experience E with respect to some class of tasks ***T*** and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E."
- Alpydin (2010) defines machine learning as " programming computers to optimize a performance criterion using example data or past experience."
- Flach (2012) avers that machine learning is about building the right models using the right features that achieve the right tasks.

Definitions (cont'd)

- Machine learning algorithm: this are the steps (procedures) followed by recognizing patterns in the data, that give it meaning. There are several algorithms used in machine learning such as regression, decision trees and so on. These will be discussed later in the course.
- Training set (data): these are the data that the algorithm uses to identify patterns, that is , what the algorithm uses to learn.
- Task: docs.microsoft (full link in references) defines this as "the type of prediction or inference being made, based on the problem or question that is being asked, and the available data"
- Label: these are the desired solutions, which will be fed together with the training data to the algorithm.
- Model: this is what is produced after running the algorithm on the training data. Models are grouped as probabilistic, logical and geometric.

Definitions (cont'd):

- It is evident from the definitions that a learning task can be divided into three distinct parameters: the task (T), performance measure (P) and the training experience (E) as defined by Mitchell (1997).
- The chess game Deep blue developed by IBM is famous for its games against then reigning champion Gary Kasparov; how was the program for the game developed in the first place? If we were to use the ML approach to develop this game we could write it as follows:
- Task (T): playing chess
- Performance measure (P): ratio of games won against competitors (this can also be measured as a percentage)
- Training Experience (E): playing games against collected grandmaster moves.
- As can be seen once a proper way of evaluating success or failure is determined then it is only experience that needs to improved in order to perfect the task.
- This approach can be applied to several tasks.

Part 3

Machine Learning Design

Design Approach

- In using a machine learning design approach it is important to follow a structured approach to solving problems through the learning experience.
- There are four steps that are involved in machine learning design as described by Mitchell (1997). These are:
  - Choosing the training experience
  - Choosing the target function
  - Choosing a representation of the target function
  - Choosing a function approximation algorithm
- Let us use a chess playing game to demonstrate how these four steps would be executed and what design issues are involved in each step.
- Let us call ours Savant, after Marilyn vos Savant the living being with the highest IQ ever recorded of 228.

1. Choosing the training experience

- Choosing the training experience is crucial to the eventual success or failure of the algorithm (learner). In choosing the training experience there are three key attributes that need to be taken into consideration. These are:
  - The type of feedback it provides based on choices made by the system
  - The kind of control given to the learner in the sequence of training examples
  - How well distributed the training examples are.
- Let us use our theoretical chess game Savant to demonstrate how these attributes would be implemented in its design.

1.1 The type of feedback it provides based on choices made by the system

- The type of feedback the training experience provides is key to the eventual success or failure of the learner.
- Consider the case of Savant; the training examples may be direct or indirect.
- Indirect training examples would consist of a dataset whereby the move sequences and eventual outcomes are availed (whether the game was won or lost). In this case it can be inferred that if the game was won then it means the moves were correct right? Well not quite! What if the game was lost but the initial moves were optimal? This is a challenge in itself for the learner.
- Direct training examples would consist of individual board states and the correct moves for each of these states.
- Based on the unknowns that indirect training examples provide it would be more desirable to have direct training examples.

1.2 The kind of control given to the learner in the sequence of training examples

- There are three options here for our learner Savant:
- She may rely on the trainer to select informative board states and to provide the correct move for each;
- She can suggest board moves she finds challenging and ask the trainer for the correct move to make;
- The trainer can be removed, thereby giving the learner complete control over different board states. This is more promising for the learner as it can experiment with different board states in the time given.

1.3 How well distributed the training examples are

- Remember Deep blue who beat world champion Gary Kasparov 4-2 the first time they met?
- Have you thought of the distribution of training examples that would have to be provided for such a system to beat someone whom many agreed was literally unbeatable?
- The importance of having training examples distributed over different situations on which the performance of the system will be evaluated can't be overemphasized.
- Bear in mind that the performance measure P for Savant is the ratio of games won vs games lost, or percentage of games won.

2. Choosing the target function

- The next step is to determine exactly what type of knowledge will be learned and how this will be used by the performance program.
- Consider the case of Savant and the chess board. In a game of chess there are several pieces, all of which make different moves on the board: king, queen, knight, bishop, castle and pawn. This means that the algorithm must select the legal moves for each piece on the board; further it must select the best move from among the legal moves.
- How is the best move chosen from the space of all legal moves?
- Obviously a function has to be formulated which will determine which is the best move among the legal moves.

2. Choosing the target function (cont'd)

- Obviously a function has to be formulated which will determine which is the best move among the legal moves.
- This function is referred to as the target function.
- In the case of Savant the target function will take as input a board state, say B, and generate as output the best move say Y (for Yes, or best move). Let us call our target function BestMove. In machine learning the notation to use in describing the target function is:
- BestMove: B → Y

2. Choosing the target function (cont'd)

- BestMove is an ideal target function. However, depending on the type of training experience available to the system it is sometimes more suitable to use an evaluation function, for example in this case one that will assign scores to different board states with the value of Y changing accordingly; this will help in determining the best move to make. Let us call this evaluation function F and the board scores R; the notation for this will now be F: B → R
- In practical situations, however, the ideal target function is hard to achieve and the learning algorithm will use an acceptable approximation which is referred to as the function approximation.
- In terms of notation, the ideal target function can be differentiated from the function approximation using the ' sign; for example if the target function is F then the function approximation will be F'.

3. Choosing a representation of the target function

- Now that the ideal target function F has been determined the next question is how to represent the function F' that our learning algorithm will learn. How will this be done?
- There are many options for the designer to use, from using tables and rules, to using different polynomial functions.
- For purposes of easy demonstration consider a given board state and what the algorithm will consider:
- In this state the algorithm will consider the number of pieces on the board from both sides, and the threats that either king is in based on the opponent pieces. This represents 6 different features or attributes of the board.

3. Choosing a representation of the target function

- This means that our function F′ can now be expressed in the form of a linear function based on the 6 attributes. Mathematically this can be expressed as:
- F′(B) = b_0 + $b_1x_1 + b_2x_2 + b_3x_3 + b_4x_4 + b_5x_5 + b_6x_6$
- Where b_0 .... b_6 are all weights to be chosen by the learner,
- And $x_1 \ldots x_6$ represent the different attributes of the board as described in the last slide.

3. Choosing a representation of the target function

- The weights b_1 .... b_6 will now be learnt by the algorithm and will ultimately determine the importance of a given attribute in the function; the weight b_0 will just provide a constant to add to the board value as it is not tied to any value of the attributes of x
- Our partial design can now be represented as follows:
- Task (T): playing chess
- Performance measure (P): ratio of games won against competitors (this can also be measured as a percentage)
- Training Experience (E): playing games against itself (we've now changed this as per the design recommendation of 1.1)
- Target function: F: B → R
- Target function representation:
- F'(B) = $b_0 + b_1x_1 + b_2x_2 + b_3x_3 + b_4x_4 + b_5x_5 + b_6x_6$

4. Choosing a function approximation algorithm

- In this step the learning function F' is determined using a set of training examples based on different board states B.
- This is done by determining different values of b against different values of x for a given board state B.
- From what we have learnt we can now describe the main elements that make up a learning system:

Learning System Components

- Mitchell (1997) describes these components as:
- Performance system: the one that solves the given performance task (T), by using the learned target function(s), for example the evaluation function F' in this lesson.
- Critic: it takes as input the history of the chess game (the example in this lesson) and produces as output a set of training examples of the target function.
- Generalizer: takes as input the training examples and produces an output hypothesis that is its estimate of the target function.
- Experiment Generator: it takes as input the current hypothesis (currently learned function) and outputs a new problem (i.e., initial board state) for the Performance System to explore. Its role is to pick new practice problems that will maximize the learning rate of the overall system

Part 4

Example Applications

ML Applications

- Machine learning applications are all around us; most times we even miss them or even assume them, as some of them are part of our daily lives.
- Who, in this digital age can claim not to know Siri by Apple and Alexa by Amazon? These digital assistants get to learn you as you keep using them and use advanced machine learning techniques to “ooh” and “aah” you as you use them.
- Amazon have also introduced Amazon Lex, which will understand your questions, detect your sentiments from your responses (yes!), and even try to complete your queries for you.
- Facebook is the most used online social media network globally. According to statista.com they had 2.91 billion users as of the fourth quarter of 2021. I can safely assume you or someone you know, is on FB as we all know it. They use ML technology to enhance your experience on the site, from facial recognition to text and video analysis.

ML Applications

- Uber also use ml algorithms to improve the efficiency of their services. For example their algorithms determine which routes will need their services more and they will send their drivers there. Their algorithms also offer drivers the best routes to take, among other functions. All this is done using the data collected from their customers, which is used to train their algorithms to be even more efficient.
- Part of the hype about iPhone is their cameras and the quality of photos (I'm sure Samsung and other top notch brands won't agree, but story for another day). Their cameras use ML technology to give images which aren't shaky for example, among other features.
- How about Microsoft? Their chatbots also use ML in their chat windows to answer customer queries.

ML Applications

- How about Google? They use ML technology to give users of their services a user friendly experience. Is their any wonder that the Google search engine is the most popular globally? In fact dictionary.com have google as a word. The search engine uses ML technology to give you the most relevant results when you search for something. The same information is then fed to their algorithms as training data, in order to constantly improve your experience.
- How about Netflix, with it's over 200 million subscribers? If you are a subscriber you have noticed that there are always movies recommended for you when you open your account; ML right there. The algorithms study what types of movies and series you like watching and make recommendations based on this.

ML Applications

- Another area in which ML has greatly helped is in medical diagnosis. ML can be used in formulating diagnosis and also recommending treatment options by studying all available data and feeding it to algorithms as training data. Currently ML is also being used to detect cancerous tissue in patients.
- ML is also used in the field of predictive analytics; it can be used to detect the legitimacy of transactions (whether genuine or fraud), real estate pricing among other areas of predictive analytics.
- ML is also used to discover learning associations. This is helpful for businesses like supermarkets; data shows that when people buy say product X, they will also buy product Y. This helps in creating an association between the two products and perhaps even have them near each other on the supermarket shelves.
- There are many other ML applications and the list will just keep growing as ML takes shape in our daily lives.

Part 5

Categories of Machine Learning

Introduction

- Machine learning systems can be classified according to the amount and type of supervision they get during training. There are four major categories:
  - Supervised learning
  - Unsupervised learning
  - Semi supervised learning
  - Reinforcement learning

1. Supervised Learning

- The training data fed to the algorithm includes the desired solutions, earlier defined as being called labels
- A typical supervised learning task is classification.
- A spam filter is a good example: it is trained with many example emails along with their class (spam or ham), and it must learn how to classify new emails.
- Another example of classification is in banks. A bank uses classification to determine credit scores for customers. This enables the bank to assess whether a customer can take a loan or not. By using different parameters such as age, income, and so on data can be fed as training data to the model and this will classify customers according to some rule; which will in turn determine whether a customer can take a loan or not. The algorithm will use a discriminant to classify those who can take a loan and those who can't. For example a discriminant can be as follows: if the customer income is greater than 150000 and savings greater than 100000 then customer is low-risk, otherwise high-risk. Low-risk and high-risk are the classes in this instance.

1. Supervised Learning (cont'd)

- Another typical task is to predict a targeted numeric value, such as the price of a car, given a set of features or attributes (mileage, age, brand, etc.) called predictors.
- This sort of task is called regression; any problem where the output is a number is referred to as regression.
- To train the system, you need to give it many examples of cars, including both their predictors and their labels (i.e.) their prices.
- To obtain the value of the car say Y, an equation is used of the linear form as describe earlier in the Savant chess game. The equation in this instance can be of a quadratic or polynomial nature. The knowns in this case will be the car attributes and by feeding training data to the algorithm the value of the weights can be determined.

2. Unsupervised Learning

- In unsupervised learning, the training data is unlabeled; this is unlike in supervised learning where mapping from input to output has correct values already provided by the supervisor.
- The only thing provided is input data, and using this regularities can be found.
- The system tries to learn without a teacher, for example, to cluster data points according to some criterion.
- A good example of this is when companies want to know more about the demographics of their customers. For instance to know which geographical location their customers are from, or some other criteria that can group their customers. Based on this the company can make business decisions. For example a bank can use this to determine where there is a need for it to open more branches based on how many customers live in that area, that have to travel distances to get to the nearest branch. They have to do this quickly before the competition lands and brings services nearer to the customers.

2. Unsupervised Learning (cont'd)

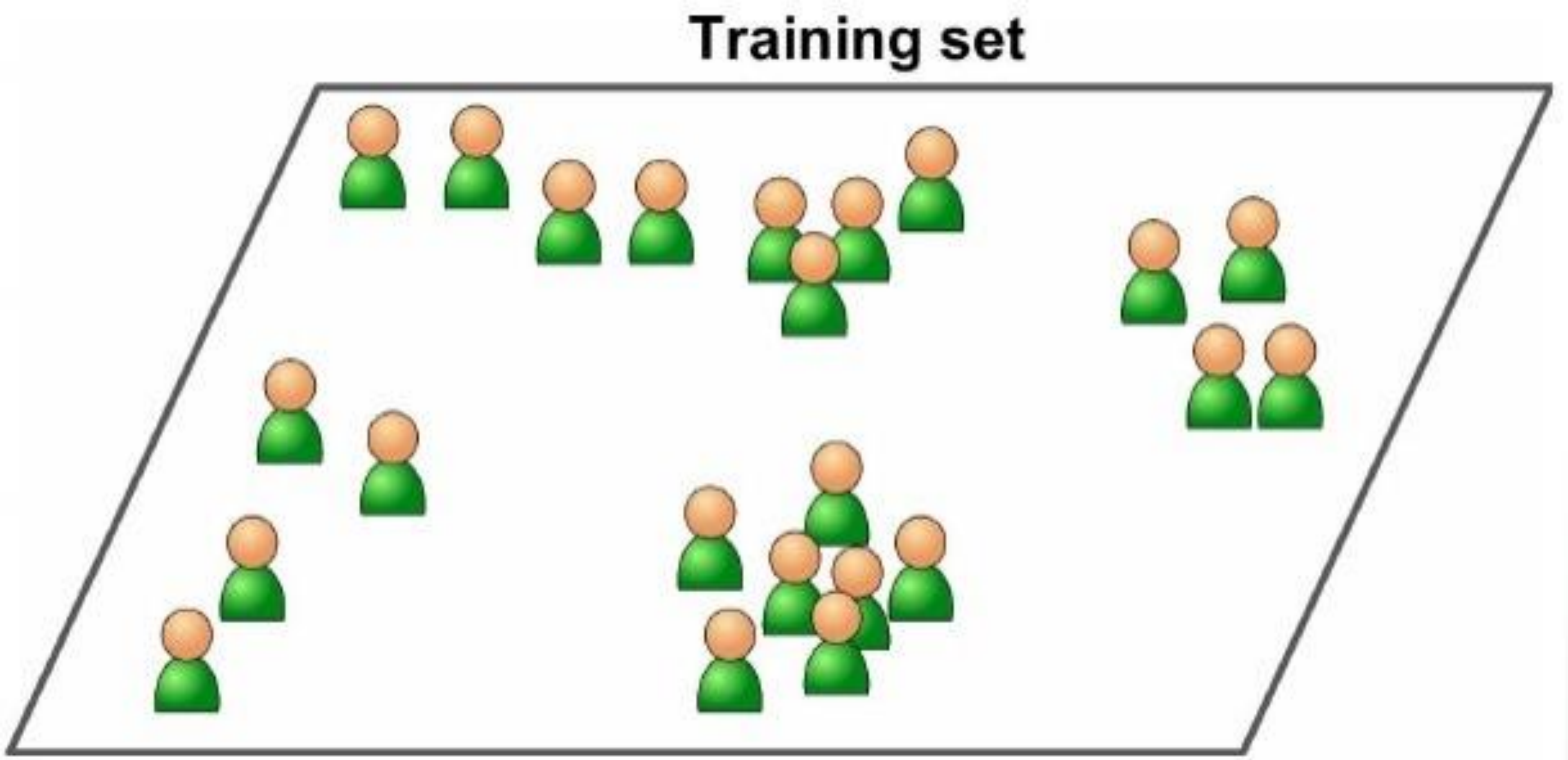

Fig 1. Example of clustering in unsupervised learning

2. Unsupervised Learning (cont'd)

- Another example of unsupervised learning is in discovering clusters of blog visitors
- For example, say you have a lot of data about your blog's visitors. You may want to run a clustering algorithm to try to detect groups of similar visitors.
- At no point do you tell the algorithm which group a visitor belong to; it finds those connections without your help.
- For example, it might notice that 20% of your blog's visitors are from Kenya who love money making books and generally read your blog in the early morning hours, while 60% are from Boston and visit during the weekends, and so on.
- This can help you target your posts for each group.

2.1. Anomaly Detection

- Another important unsupervised task in anomaly detection, for example:
  - Detecting unusual credit card transactions to prevent fraud
  - Catching manufacturing defects
  - Automatically removing outliers from a dataset before feeding it to another learning algorithm.
- The system is trained with normal instances, and when it sees a new instance it can tell whether it looks like a normal one or whether it is likely an anomaly.

2.1. Anomaly Detection

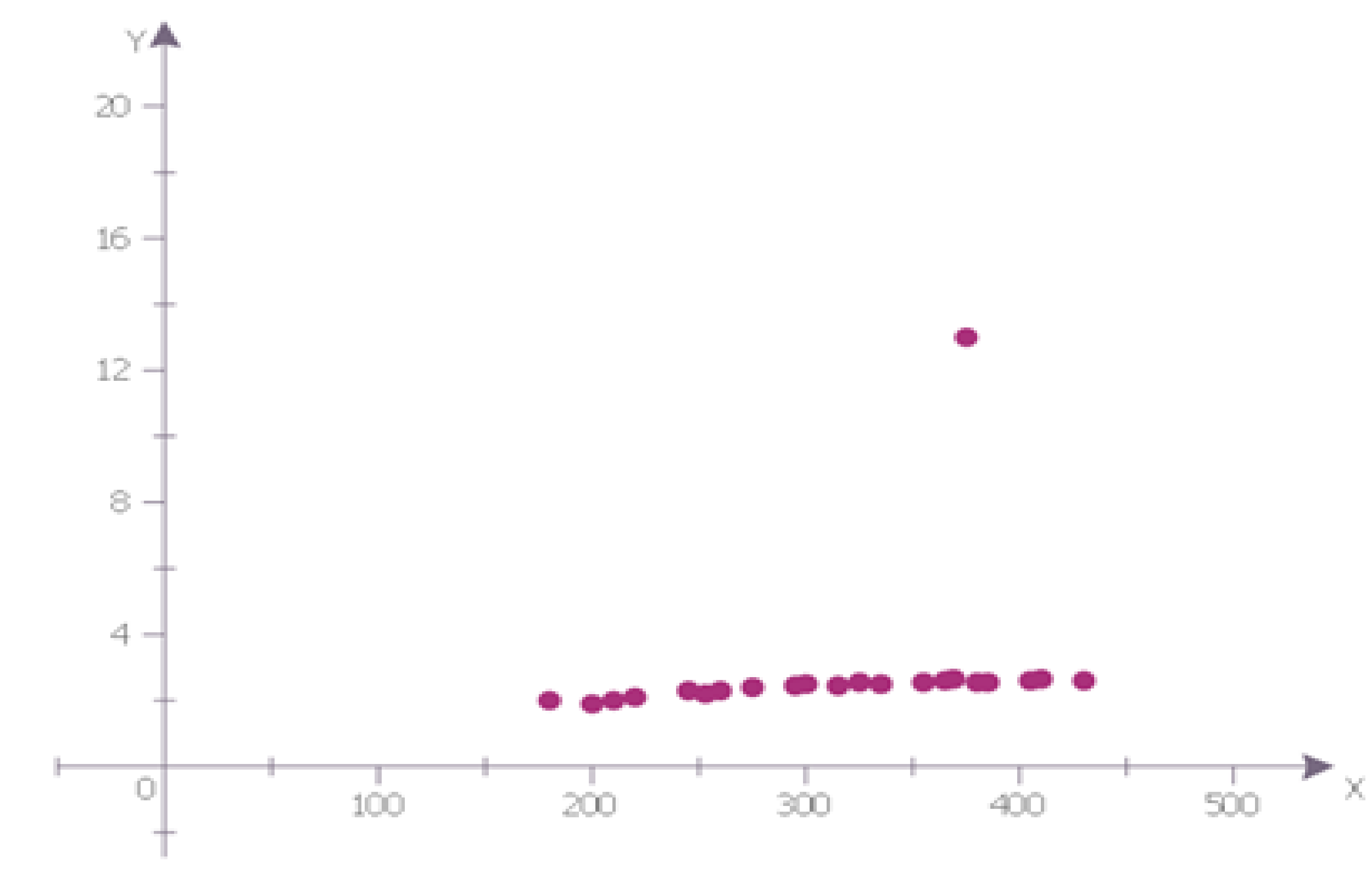

Fig 2. Example of anomaly detection (outlier) (medium.com)

2.2. Association Rule Learning

- Another common unsupervised task is association rule learning, in which the goal is to dig into large amounts of data and discover interesting relations between attributes.
- For example, suppose you own a supermarket. Running an association rule on you sales logs may reveal that people who purchase chicken and sausages also tend to buy ice cream. Thus you may want to place these items close to each other.

3. Semi-supervised Learning

- Some algorithms can deal with partially labelled training data.
- Usually, only a small proportion is labelled, and the bulk of the data is unlabeled.
- This is called semi supervised learning; it is a cross between supervised and unsupervised learning.
- A good example is Google photos; it recognizes when people appear in more than one photo and takes note (this is the unsupervised part utilizing the clustering technique). Unfortunately even though it has clustered the individual images it still doesn’t know who they are, and it’s up to you to identify them (the supervised part). Once you have done this Google photos will tell you who is who in every photo.

4. Reinforcement Learning

- Reinforcement learning is quite different from the others we've seen so far.
- The learning system, called an agent in this context, can observe the environment, select and perform actions, and get rewards in return (or penalties in the form of negative rewards).
- The agent must then learn by itself what is the best strategy, called a policy, to get the most reward over time.
- A policy defines what action the agent should choose when it is in a given situation.
- Many robots implement reinforcement learning to learn how to walk.

Artificial Intelligence (AI) or Machine Learning?

- As a learner you are probably wondering whether what has been covered in this lesson is ML or AI?
- Well admittedly in many scenarios the two are used interchangeably.
- However, let’s hear from some authority figures:
- An “intelligent” computer uses AI to think like a human and perform tasks on its own. Machine learning is how a computer system develops its intelligence. (https://azure.microsoft.com/en-us/overview/artificial-intelligence-ai-vs-machine-learning/#introduction)

Artificial Intelligence (AI) or Machine Learning?

- AI solves tasks that require human intelligence while ML is a subset of artificial intelligence that solves specific tasks by learning from data and making predictions. This means that all machine learning is AI, but not all AI is machine learning. (https://www.freecodecamp.org/news/ai-vs-ml-whats-the-difference/)
- See also https://www.techrepublic.com/article/understanding-the-differences-between-ai-machine-learning-and-deep-learning/ and https://www.geeksforgeeks.org/difference-between-machine-learning-and-artificial-intelligence/ for more distinction between the two.

Summary

- A simple program cant give meaning to all the data in the digisphere. There is need for ML techniques to extract the data and give meaning to it.
- A learning task can be divided into three distinct parameters: the task (T), performance measure (P) and the training experience (E).
- There are four steps that are involved in machine learning design as described by Mitchell (1997). These are: choosing the training experience, choosing the target function, choosing a representation of the target function, and choosing a function approximation algorithm.
- Machine learning systems can be classified according to the amount and type of supervision they get during training. There are four major categories: Supervised learning, unsupervised learning, semi supervised learning, and reinforcement learning

References

- Alpaydin, E. (2010). *Introduction to machine learning*. MIT Press.
- Flach, P. A. (2017). *Machine learning: The art and science of algorithms that make sense of data*. Cambridge University Press.
- Luisquintanilla. (n.d.). *Machine learning tasks - ML.NET*. Machine learning tasks - ML.NET | Microsoft Docs. Retrieved March 16, 2022, from https://docs.microsoft.com/en-us/dotnet/machine-learning/resources/tasks#:~:text=A%20machine%20learning%20task%20is,groups%20data%20according%20to%20similarity.
- *Machine learning: 6 real-world examples*. Salesforce EMEA Blog. (n.d.). Retrieved March 16, 2022, from https://www.salesforce.com/eu/blog/2020/06/real-world-examples-of-machine-learning.html
- Mitchell, T. M. (1997). *Machine learning*. McGraw-Hill.
- Published by Statista Research Department, & 14, F. (2022, February 14). *Facebook mau worldwide 2021*. Statista. Retrieved March 16, 2022, from https://www.statista.com/statistics/264810/number-of-monthly-active-facebook-users-worldwide/#:~:text=With%20roughly%202.91%20billion%20monthly,used%20online%20social%20network%20worldwide.

References (cont'd)

- Shalev-Shwartz, S., & Ben-David, S. (2021). *Understanding machine learning: From theory to algorithms*. Cambridge University Press.
- VarunBhagat, & VarunBhagat. (2021, December 22). *7 real-world examples of machine learning in current times*. Data Science Central. Retrieved March 16, 2022, from https://www.datasciencecentral.com/7-real-world-examples-of-machine-learning-in-current-times/
- Ved, M. (2018, April 6). *Outlier detection and anomaly detection with machine learning*. Medium. Retrieved March 16, 2022, from https://medium.com/@mehulved1503/outlier-detection-and-anomaly-detection-with-machine-learning-caa96b34b7f6
- Wikimedia Foundation. (2022, March 3). *Deep Blue (chess computer)*. Wikipedia. Retrieved March 16, 2022, from https://en.wikipedia.org/wiki/Deep_Blue_(chess_computer)