

Book By Francisco Bettencourt

Used for AI Risk Certification Exam

**Contents**

[Module 1 - AI and Risk – Introduction and Overview 6](#_Toc175848044)

[1.0 Classical AI 7](#_Toc175848045)

[**1.1 Specific versus General AI** 8](#_Toc175848046)

[**1.2 Good Old-Fashioned AI (GOFAI) and Classical AI** 10](#_Toc175848047)

[**1.3 Simple Reinforced Learning** 12](#_Toc175848048)

[**1.4 Lookahead** 13](#_Toc175848049)

[**1.5 Search** 14](#_Toc175848050)

[**1.6 Recursion** 16](#_Toc175848051)

[**1.7 Recursive Adversarial Tree Search** 17](#_Toc175848052)

[**1.8 Complexity, heuristics and Refinement by Reinforcing Learning** 18](#_Toc175848053)

[**1.9 Limits of Classical Artificial Intelligence** 20](#_Toc175848054)

[**2.0 Neural Networks** 22](#_Toc175848055)

[**2.1 Artificial Neurons** 23](#_Toc175848056)

[**2.2 Connectionism and its Early Challenges** 26](#_Toc175848057)

[**2.3 Deep Learning Proves its Potential** 28](#_Toc175848058)

[**2.4 Deep Learning Beats Symbolic AI at its Own Game** 29](#_Toc175848059)

[**2.5 The inscrutability of Deep Learning** 31](#_Toc175848060)

[**2.6 The Dwan of Artificial General Intelligence?** 33](#_Toc175848061)

[**3.0 Machine Learning and its Risks** 35](#_Toc175848062)

[**3.1 Four Types of Machine Learning** 35](#_Toc175848063)

[**3.2 Examples of Unsupervised Learning (Principal Component Analysis)** 37](#_Toc175848064)

[**3.3 Risk of Inscrutability** 40](#_Toc175848065)

[**3.4 Risk of Over Reliance** 41](#_Toc175848066)

[**3.5 Risk To individuals, Organizations and Society** 42](#_Toc175848067)

[**Questions and Answers Module 1 from GARP** 44](#_Toc175848068)

[Module 2 – Tools and Techniques 46](#_Toc175848069)

[**1.0** **Machine Learning** 47](#_Toc175848070)

[**1.1.1 ML, Classical Statistics and Econometrics** 48](#_Toc175848071)

[**1.2 Four Types of Machine Learning** 50](#_Toc175848072)

[**1.3 Exploratory Data Analysis** 52](#_Toc175848073)

[**1.3.1 Data Collection and Preparation** 53](#_Toc175848074)

[**1.3.2 Data Cleaning** 56](#_Toc175848075)

[**1.3.3 Data Visualization** 58](#_Toc175848076)

[**1.3.4 Feature Extracting** 61](#_Toc175848077)

[**1.3.5 Data Scaling** 64](#_Toc175848078)

[**1.3.6 Data Transformation** 66](#_Toc175848079)

[**1.4 Dimensionality Reduction Techniques** 67](#_Toc175848080)

[**1.4.1 Principal Component Analysis** 67](#_Toc175848081)

[**1.5 Training Validation and Testing** 71](#_Toc175848082)

[**1.5.1 Sample Splitting and Preparation** 72](#_Toc175848083)

[**1.6 Software for Machine Learning** 73](#_Toc175848084)

[**Questions and Answers Tools and Techniques – Module 2 GARP** 75](#_Toc175848085)

[**2.0 Unsupervised Learning** 80](#_Toc175848086)

[**2.1 K-Means Clustering Algorithm** 85](#_Toc175848087)

[**2.1.1 Performance Measurement for K-Means** 91](#_Toc175848088)

[**2.1.2 Selecting the Starting Positions of the Centroids** 91](#_Toc175848089)

[**2.1.3 Selection of K** 92](#_Toc175848090)

[**2.1.4 Selection of K Example** 93](#_Toc175848091)

[**2.1.5 Advantages of K Means** 94](#_Toc175848092)

[**2.1.6 Problems with K Means** 94](#_Toc175848093)

[**2.1.7 Fuzzy K Means** 95](#_Toc175848094)

[**2.2 Hierarchical Clustering** 96](#_Toc175848095)

[**2.3 Density Based Clustering** 99](#_Toc175848096)

[**Extra 2.A Different Distance Measurement** 100](#_Toc175848097)

[**Extra 2.B Silhouette Method** 102](#_Toc175848098)

[**Extra 2.C K-Means Clustering Example** 103](#_Toc175848099)

[**Questions and Answers Module 2 Chapter 2 from GARP** 107](#_Toc175848100)

[**3.0 Supervised Learning for Numerical Data** 112](#_Toc175848101)

[**3.1.1. Simple Linear Regression** 118](#_Toc175848102)

[**3.1.2. Multiple Linear Regression** 120](#_Toc175848103)

[**3.1.3. Potential problems with Regressions** 126](#_Toc175848104)

[**3.1.4. Stepwise Regression Procedures** 131](#_Toc175848105)

[**3.2. Classification Problems** 134](#_Toc175848106)

[**3.2.1. Logistic Regression** 135](#_Toc175848107)

[**3.2.2. Other Types of limited Dependent Variable Models** 137](#_Toc175848108)

[**3.3 Linear Discriminant Analysis** 138](#_Toc175848109)

[**Appendix 3.A The Heckman 2 stage Procedure** 141](#_Toc175848110)

[**Appendix 3.B Fisher Discriminant Analysis** 142](#_Toc175848111)

[**Appendix 3.C Linear Discriminant Analysis Example** 144](#_Toc175848112)

[**Appendix A -Q&A Module 1 from CHAT GPT** 153](#_Toc175848113)

[**Appendix B -Q&A Module 2 from CHAT GPT** 173](#_Toc175848114)

[**Appendix B -Q&A Module 2 Chapter 2 from CHAT GPT** 201](#_Toc175848115)

# Module 1 - AI and Risk – Introduction and Overview

Key topics Being covered on this chapter include:

How to explain some central principles around classic AI, including search methodologies and recursion.

Describe, at a high level, how reinforcement learning works.

Describe, at a high level, how neural networks work, and how they differ from classical AI systems.

Articulate the potential and limitations of deep learning.

Identify key breakthroughs leading to advances in AI and ML.

Compare and contrast reinforcement, supervised, and unsupervised learning, and identify practical applications for each technique.

Discuss Risks associated with inscrutability in AI and ML.

Discuss Risks associated with over-reliance on AI systems

Discuss ways in which AI exposes individuals, organizations and society to risk.

## Classical AI

Intelligence or reason is a quality that distinguishes humans from other species. There has been always the speculation about the possibility of duplicating the power of human reason with artificial technology. This concept began to seem feasible after the industrial revolution, where Charles Babbage conceived a mechanical, gear-driven “difference engine” that could replace error-prone human calculations in the efficient production of mathematical tables.

Later, the analytics engine is widely seen as anticipating the digital computer, through bringing it to successful completion would be a huge feat of engineering , even with moder technology.

The Crucial breakthrough occurred when the digital computer emerged as the brainchild of Alan Turing in 1936, although his “computing machine” was a theoretical invention.

It was designed to prove fundamental results about mathematics and what we now call the theory of computation. Such machine would be able to follow any given recipe for a calculation that we were able to devise, and in that sense would be a universal programmable computer.

Turing presented his visionary ideas and thoughts about the tremendous potential power of intelligent machines on his paper, “Computing machinery and intelligence” dated 1950. This Centers on an idea of an imitation game, a.k.a Turing Test, in which a computer proves itself to be intelligent if it can generate textual conversations of a quality indistinguishable from that of an intelligent human, raging over any topic chose by the Human “Interrogator”.

This is highly controversial philosophically but has become more salient recently due to chatbots based on Large Language Models (such as ChatGPT). These ones have provided, for the first time, examples of computer programs that are relatively plausible contenders to achieve such high levels off intelligence.

## **1.1 Specific versus General AI**

Media discussion and futuristic speculation around AI, is often concerned with the question of whether an artificial system could surpass human general intelligence. This Concern originates from the legacy of Turing Test, but it would probably reflect a natural concern about challenges to our supremacy in reason and understanding, which are abilities that are so bounded to hour species identity.

Such concepts of “fear” associated, often linked to the existential risks arising from Artificial General Intelligence (AGI). It is far less threatening to consider that AI can surpass human reasoning in specific tasks, such as calculating mathematical tables or generating complex statistics. But, it is this type of Machine, the one designed to be intelligent at solving specific problems, that will mostly concern us in what follows.

Some will count a system as intelligent, if this one is provided of the general ability to be intelligent. It has become commonplace to talk of intelligent or “Smart” systems that are relatively limited. Even Turing’s legacy does not point very heavily towards a purely general interpretation of AI.

In 1950, on his paper, the idea was to demonstrate the possibility of such a computer system that could not be reasonably denied as being intelligent. In this sense, it makes clear sense to consider a system that was based on conversational principles, that would allow to compare in a versatile way against the reasoning and skills of an Human, as a sort of conceptual existence proof of that possibility.

But once we have been convinced that there is no objection in principal to artificial intelligence, it seems odd to deny that a system can be intelligent in addressing specific tasks, without having more general competence.

## **1.2 Good Old-Fashioned AI (GOFAI) and Classical AI**

This includes techniques that try to automate the sort of reasoning we do.

The term artificial intelligence was coined in 1956 by John McCarthy of Dartmouth College. Echoing Turing’s notion of universality, this has been the conjecture that every aspect of learning or any feature of intelligence, can in principle be so precisely described that a machine can simulate it. The paradigm of intelligence was accordingly explicit, precise processing, as typified by symbolic logic or other formal rule-governed system.

In 1936, Turing had emphasized their ability to mimic logical proofs, constructed step by step from a given set of “Axioms” and systematically applying such rules. Following in this tradition, in 1955 Allen Newell, Herbert Simon and Cliff Shaw (Rand Corporation) started developing their Logic Theorist Program, designed to generate proofs of propositional logic, the most simple and fundamental form of Modern Symbolic Logic. Their work was used to prove the Logistic Thesis, that all of mathematics could be deducted from suitable axioms by pure logic, so it was an excellent choice of high-profile application of machine intelligence.

Another paradigm aspiration of AI was to create a program capable of playing “intellectual” chess at a high level.

Alan Turing have tried to implement this, but his ideas never left the paper due to their engineering limitations at the time. Later, in 1997 Deep Blue own against Garry Kasparov due ti more powerful machines and sophisticated techniques.

Quite apart from their intrinsic interest and their promotional value in raising public awareness of the possibility of AI, such intellectual games provided an attractive testbed for the development of AI for at least three reasons.

**First,** they are relatively Simple, compared to the complexities of day-to-day life, and hence could plausibly be tackled even when computing resources would have been severely stretched in even recording the details of a physical environment, let alone taking on the immensely complicated challenge of reasoning about it.

**Secondly,** such games are straightforwardly rule-governed, with clear and explicit regulations, regarding the setup of the game, how the pieces are permitted to move, the effect of such moves on the boar position and criteria for winning, losing or drawing.

**Thirdly,** they are competitive, and we already have the idea that players can be rated according to their relative skill level.

Hence, such games, lend themselves to very naturally to the assessment of that skill, to that it is straightforward to judge researchers progress as they endeavor to produce a program capable of competing at the higher levels.

## **1.3 Simple Reinforced Learning**

Classic Ai works in a way that we can understand well, using logical reasoning, clear representation of data, and explicit calculation. This differs quite well from neural networks; it is vital to be aware of the differences to understand the majority of risks that contemporary AI involves

Reinforcement Learning, plays a huge role in Contemporary AI. Reinforcement learning is where an agent, that can be a human or AI, that by applying different actions, that generate an outcome, and by observation experience tries to understand which works betters.

Successful actions are positively reinforced, which are more likely to occur in the future, and vice versa. The likelihood of such action is contained in some type of numerical expression.

In moder machine learning the role of this weights are hard to understand.

So, from the game example, if a computer that is based on supervisory learning plays a random opponent, will eventually became much better playing, but his capacity of learning will be biased by the fact that the player sometimes losses games it should own, meaning that the computer is getting positively reinforced on incorrect answers.

Now, if the player is also a reinforcement learning agent, this means that will always try to achieve the Successful action, and that his actions, since is being successful, are being constantly reinforced. By this, the computer, playing against a perfect player, can in a much faster pace, improve is capacity to play, since it is being positively reinforced on Successful Actions.

The later, means that the computer as achieved a perfect capacity of prediction , while in the random scenario, the computer chances of winning are x times higher, but since learns from mistakes, it is not perfect.

## **1.4 Lookahead**

The reinforcement learning was playing noticeable but not on an intelligent way. Searching amid all possibilities, and search/look ahead for opportunities.

Binary search, the possibility range is divided by 2. (Decision Three).

Breadth first search explores all the routes in parallel, eliminating as we go, those that can be the sub-optimal.

Depth-Search we explore each route in turn to the end (while avoiding loops- these are the options we crossed out)

A\* uses a heuristic – here an estimate of time still to travel, based on crow-flies distance allows us to assess how far a station is from oxford, in this case choosing the one closest to Oxford .

Lookahead become essential when our aim is to form a multistep plan to achieve some goal, within a context where there is a very large number of possible situations overall, but with a relatively constrained and predictable range of options in any particular situation.

We follow a sequence of implications with the idea that eventually we will either find a contradiction, which lead us to rewind/backtrack the path chose, or we will generate a complete solution. Focus on the promising of using a strategic planning approach, whereby we try to identify what short-term outcomes would usefully contribute to the completion of our task.

## **1.5 Search**

Working through various options, we are conducting a search through the possible outcomes to find one that will best achieve our aims. Generating sets of possible outcomes and searching through them is one of the most fundamental techniques in AI.

In a binary search, the space is progressively divided in two until the desired target has been found. Whenever a guess is unsuccessful, the “Search Space” is reduced by at least at half, removing the part that has been ruled out by the user’s response. Assuming that the answers are consistent the x number of times needed will equal to:

A more complicated way is attempting to find the optimal route from one city to another by taking an appropriate sequence of journeys, with no obvious best combination to reach our destination.

Disadvantages of using breadth-first search is that it is inefficient and expensive in memory, whereas using depth-first search may waste time focusing on poor options initially while neglecting short and straightforward paths that the breadth- first search would have found far more quickly.

Better than the above is to employ methods of heuristic search, where we take advantage of some additional information, to help us decide which options are worth exploring first. The best-known example is the Technique A\* algorithm, in which the locations are kept ordered according to the sum of the distance travelled so far and an optimistic estimate of the distance still to go, and exploration of ongoing routes always starts from the location that currently has the shortest sum.

Heuristics techniques such as the one presented, can indeed help in facilitating many search problems, but unfortunately in more complex or demanding cases the search space grows so enormously as the relevant number of possible inputs increases.

Leading to a combinatorial explosion as the relevant number of possible choices gest multiplied together, making the certainty of a solution unfeasible.

## **1.6 Recursion**

Fundamental to computational science and base for classical Artificial Intelligence.

Divides the complex task into a simple task… Its convenient to be the same task category. A recursive function is a function that calls itself. Reducing the complicated task by reducing them to simple examples of them selves

Tower of Hanoi is one of the most famous and elegant illustrations of the recursive technique, whereby a complex problem is solved by being progressively reduced to simpler instances of the same kind of problem.

This simple line of reasoning provides a complete solution to the tower of Hanoi, dividing every problem of moving a pile of n disks into two smaller problems of moving pile of n-1 disks (plus one move of disk n) so that in the end everything is reduced down to single disk movements. Moving the entire pile of n disks by this method is given by:

## **1.7 Recursive Adversarial Tree Search**

Recursion can also be applied to solve the relatively complex problem of playing an adversarial game, by combining lookahead and search.

On TIC TAC TOE, is crucial to devise a function that can assess the value of any position to the player whose turn it is to move in that position. Red, should always play the move that gives Blue the lowest possible value in the remaining n-1 positions.

Adversarial search is where red is supposed to look for whichever move is least advantageous for her opponent.

The above is only true if we have an oracle, infallible method. In the real world, what we should do is then repeat the same sort of reasoning at the next level, as to consider a possible, suboptimal, continuation from the previous position.

This strategy should be carried out until where Blue has an immediate winning move.

Consequences of this line of thinking:

If we can evaluate any n-1 position, then we can evaluate a n position.

This pattern continues, with the upshot that if we can evaluate any 0-positon (i.e. position with no remaining moves to be played), then we can evaluate any 1-position, and so on until 9- position

Apart from this, we need to be able to recognize when a game has ended owing to completion of a line (yielding a position of -1 to the player whose turn it would otherwise be).

## **1.8 Complexity, heuristics and Refinement by Reinforcing Learning**

The previous example, tic tac toe, that is an assessment algorithm that is capable of exhaustively analyzing all the possible lines of play in a game, implicating creating a “search tree”, and keeping track of those positions value can be implemented so simply (less than 25 lines of code).

There is a catch to this, that is the curse of exponential complexity that leads to a combinatorial explosion. For more complex games, the number of possibilities per Move is just so large that will take forever with today’s computational capacity.

As Arthur Samuel IBM, a practicable checkers program could not possibly work by exhaustive search through all possible combinations of moves.

At any point in the game position, it should limit its exhaustive search to n moves and then to assess the positions reached through this analysis, and then select which path is best to follow) by using heuristics. In other words, calculable criteria or rules of thumb that can usefully guide the choice of position to aim for, even though they cannot be guaranteed to reach an optimal solution.

By assuming a specific number of criteria, then we can give them some weights, and at each point, to compare one position with another overall, we should calculate the value for each position by the following formula

Choosing a move in a particular position now becomes a matter of examining the positions that would arise from all the analyzed possible sequences of n moves in that position.

If two moves provide the same value, we should choose randomly between them, hence yield the twin advantage of making the program less predictable and providing a more-varied basis for learning (the program).

Human judgment, provided by practical experience with the game, is involved here in identifying plausible criteria for assessing positions, which indeed might seem fairly straightforward for an expert player. But, in this case, how many relative weight should we provided to each criteria?

The question above is far more difficult to assess, but there is no need to rely on Human judgement when selecting the crucial weights that feed into our move-selection algorithm, for this is where reinforcement learning can play a role.

By continuously play interactively, the weights in our current strategy are progressively refined ( by simulated evolution) and we hope to ultimately achieve something close to an optimal set of weight, and thus the best possible strategy of this kind.

Depending on the game and suitable choice of criteria, this sort of machine learning technique can enable a program to learn to play better than its designer.

## **1.9 Limits of Classical Artificial Intelligence**

Based mainly on the simple context of familiar rule-governed games.

In less well-disciplined contexts, however, classical AI techniques struggled to fulfil their apparent early promise. One serious difficulty was how to represent the characteristics and relationships of things in the world, and especially the rules that govern their behavior.

Even human experts find it very hard do elucidate the implicit background assumptions that guide their judgements, and spelling out our ordinary common-sense understanding of things proves to be extraordinary difficult.

A related issues was the so called “frame problem”, keeping track of which aspects of a situation change when some action is performed, but also which aspects stay the same.

Codifying these things in detail added yet more fuel to the sort of combinatorial explosion that we have already mentioned, which quickly, resulted when attempts were made to apply general purpose search mechanisms to classical AI problems of any serious complexity.

This becomes even worse if the data used is ambiguous or uncertain, requiring yet more complex calculation if probabilities were to be considered.

Real-time data about physical things was inevitably error-prone, with computer vision systems relying on tailor mad algorithms, first to identify such things as edges, shapes, textures and colors, and then to synthesize these features into the representation of similar objects.

In response to such limitations the focus of much classical AI research moved during 1980s toward so called expert systems, which were designed to capture knowledge within specific domains, often using logic programming as general purpose reasoning mechanism.

These systems, despite being very successful, their functionalities could not easily be generalized beyond those specific domains without hitting the same old problems.

AI were notoriously brittle, failing in unexpected ways when applied beyond their familiar narrow boundaries, or to situations that had not been explicitly foreseen by their designers.

With this, progress continued everything that could be feasibly assessed yielding impressive results, while the dream of Alan Turing’s of general artificial intelligence, was likely to remain unfulfilled, at least for the foreseeable future.

## **2.0 Neural Networks**

Alan Turing had discussed AI around 1941, but what is often considered the first published contribution to modern AI development appeared son after in the Bulletin of Mathematical Biology of 1943 by Warren McCulloch and Walter Pitts.

This paper intended as a contribution to the theoretical neurophysiology , describing the activity of neurons in the brain, and proposed that the behavior of these neurons could be analyzed in terms of propositional Logic.

Ironically in view of the late contrast that would be drawn between logical and neural approaches the paper task of neural nets as equivalents as Turing’s machines, and hence as limited by Turing Computability.

Donald Hebb’s 1949, suggested a way in which associative learning could take place within networks of neurons, based on “neurophysiological postulate”, in which states that One cell by proximity of another, or by firing another, will suffer such a change that will increase its efficiency.

Frank Rosenblatt later took this further, proposing a specific computational model for how the brain learns and stores information in which, rather than treating memories as encoded representations of experiences, he instead favored the Hebbian approach.

Rejecting any symbolic or algorithmic approach, he accordingly formulated his new model, the Perceptron, in terms of probability theory rather than symbolic logic.

## **2.1 Artificial Neurons**

Rosenblatt’s Perceptron led to a standard model of an Artificial “neuron”, which takes several numerical inputs ( and outputs a single value V. Each input is multiplied by some weight , and these are added together with some constant bias term as to make the net input. This is later fed into an activation function to generate the output value, such as:

The activation function is chosen to be non-linear, and often approximates to a step function that gets triggered when the net input is greater than some threshold, making the neuron’s output an “All or nothing”. This non-linearity is crucial if a network is to be able to learn non-linear behavior.

In recent years, artificial neurons of this general kind have been powerfully linked together in “deep” networks that have multiple additional layers between the input and the output layer.

From figure 1.6, the greens are the input layers, the blues are the hidden layers, and the red ones are the output layers.

Each neuron in the input layer is given a specific numeric level of activation, in such a way that the overall pattern of activations represents the input data.

For instance, each neuron’s activation might store the color value of one of the pixels in an image. In this example, every neuron in the input layer is connected to every neuron in the first hidden layer (a- layer), so that the level of activation of the a-neurons will depend on their activation function as applied to the net input that they receive from the input layer.

In the same way, every neuron on the second hidden layer (b-layer) will depend on their activation function as applied to the net input that they receive from the a-layer, and so on.

Typically, the activation function will be consistent across the neurons, but the weights of the individual connections will vary, thus influencing the propagation of activity through the network from the input layer, through the hidden layers until the output layer.

It is the changing in weights that represents the learning of a network. In a successful trained network, the weights will have evolved in such a way that the activation of the output neurons does indeed provide the desired response to the relevant input.

A screenshot of a computer

Description automatically generated

## **2.2 Connectionism and its Early Challenges**

Rosenblatt’s work provided widespread interest in artificial neural networks and an appreciation of some of their particular strengths as learning mechanisms.

In 1969, Marvin Minsky and Seymour Papert published a fierce critique in their book Perceptron’s, which undermined the enthusiasm.

Perceptron’s were incapable of learning some simple logical functions, thus apparently wrecking any prospect that they might provide a route toward “intelligent” information processing of any complexity.

The above is through for neural networks containing a single layer, and that more complex functions could be learned by arranging artificial neurons into multiple layers, hence denominated Deep Neural Networks.

Adding additional layers complicated the process of learning until the Back Propagation on the 1980s

Th Output should match the input when the activation of the function is closets to 1 (maximal activation) and vice versa, when close to 0 minimal activation. However, if the actual pattern of activation is quite different then learning is required, so we should calculate how much a small change in the weights of the final output layer would contribute to improving the match.

Back propagation is then the process of working back through the layers, doing a similar calculation for all the other weights in the network.

Once this has been done, the weights are now adjusted by a small amount, in the direction that would bring improvement to the model capability of forecasting.

This process should be repeated thousands or millions of times so that the network learns by iteratively adjusting its weights.

The Discovery and success of back propagation led to a resurgence of interest in what became known as “connectionism”, in 1986

This show that layers of simple interacting neurons, could achieve learning of complex, cognitively relevant functions.

There were 4 main reasons for the comeback being that:

**1st** is that this is biologically inspired and could provide insights on humans think.

**2nd** it seemed to learn in the same we do, through association and feedback in response to success or failure.

**3rd** this was really general and could cover a bunch of domains.

**4th** the storage of the learned information, rather than being explicitly represented, was obscurely distributed through the network of weights, which made the learning process more robust in response to “noisy” of ambiguous data, better able to generalize and less liable to break down entirely as the quality of data declines.

In the 1990s the enthusiasm declined due to the impossibility of replicating in practical terms the theoretical concepts. Such concepts would come later as the computational capacity increased.

## **2.3 Deep Learning Proves its Potential**

In 1998 Yann LeCun, publish work on a deep network with several “convolutional” layers, each of which has the effect of applying a local filter, Kernel, repeatedly to points across the grid, which in this case is a grid of pixel values in a greyscale digitized image.

The filter consists of a small matrix of numerical weights, and each of these weights Is multiplied by the pixel activation value at its position, with the sum of these products providing the corresponding activation value in the convolutional layer (at the filters central point).

The weights at the beginning are set as random, and then learn in the same sort of way as other weights on the network. The virtue of such a convolutional layer is to enable efficient identification of local features in an image, which can then feed into the remainder of the network.

The model contained 8 layers, starting with the input layer (containing the pixels from the image), then two sequences of convolutional layers, followed by a “pooling” layer (which reduces the dimension of “feature map” by dividing it into 2X2 squares and averaging, then two further layers, and finally the output layer that would classify the image.

In 2012 AlexNet own the ImageNet Large Scale Visual recognition challenge with only 15% error margin. The Network was nearly 100 times bigger than the Yann LeCun one, holding around 900.000 neurons.

The computational power required had been obtained due to the high demand for high quality animated computer games, that required powerful graphic processing units, that turned out to be great for huge matrix operations that are required by neural networks.

## **2.4 Deep Learning Beats Symbolic AI at its Own Game**

Another breakthrough arrived in 2013, when the London Company Deep Mind announce its success in programming a deep convolutional network to learn to play vintage Atari video games from the 1970s, which outperforms vastly better than an expert human.

The major breakthrough was that the system was not provided with any information regarding the game’s goal, the information on the screen or the effects of user action (e.g. pressing buttons). It had to learn to do entirely on the basis of knowing that a certain number of distinct actions were available and trying these out in response to pixel information about the changing images and the game score. Everything else was done by deep reinforcement learning based on the score feedback, so in a sense, the system was teaching itself how to play from scratch.

In 2014 DeepMind was acquired by google and soon after created AlphaGo, a system capable of beating Go world champions, that is a game that until the moments was considered to subtle and complicated for computer algorithms to master in the foreseeable future.

Later, Alpha zero was able to teach it self other games, with residual user inputs, such as how to identify a win, how pieces can actually move, general rules of thumb and had efficient recursive adversarial tree searching built into it.

In contrast to Deep Blue and other traditional AI programs, AlphaZero learn how to play entirely on its own without any input from human experts or game databases. A few hours of self-training allow it to defeat the Stockfish champion program.

In this way, AlphaZero was altogether a impressive system, teaching itself new skills to an even higher level than human ingenuity had been able to achieve, and thus representing massive and potentially frightening progress towards artificial General intelligence.

## **2.5 The inscrutability of Deep Learning**

Part of the promise and threat of deep learning lies precisely in its ability to represent all kinds of information in ways that it works out for itself in response to training data and feedback. This information is stored implicitly within the weights of a neural network that consists of layers of artificial neurons.

The input layer of the network is setup to reflect the specific problem case. The output layer is set up to signal the corresponding solution. But for a deep, multilayer system, the activation of each neuron depends on the input received from the neurons to which is connected on the immediate previous layer ((possible involving convolutions or other kinds of local processing).

These inputs depend both on the level of activation of those previous neurons, but also on the weights given to the relevant connections. These weights are adjusted during the reinforcement learning process (by back propagation or Refinements thereof), which typically involves going iteratively through the training data, assessing the results outputs, and gradually refining the relevant weights until a sufficient match between inputs and outputs has been matched.

Neither the weights nor the roles of the neurons in the intermediate layers are predetermined when the learning process starts.

By the end, the immensely complex pattern of weights implicitly represents what the network has learned, but in a way that an unaided human will find impossible to interpret, and whose behavior they will be able to predict only be experience.

Humans should not be misled to suppose that the machine themselves “understand” what they are doing in any reflective way, and not only because they are completely non-sentient, and hence, have no awareness or conscious understanding of anything).

In conspicuous contrast to Classical AI techniques, their way of working is far more closely analogous to our own unconscious pattern-recognition than it is to how we think when explicitly calculating or reasoning about something.

A proper analogy is to a chess grandmaster, that without knowing empirically how to fundament his decision on a given position, will take it. The same with the system. The grandmaster also needs to be able to understand when a tactical combination is imminent. But as mentioned before, AlphaZero, can perform explicit calculation, exploring options down the “game tree” of branching possibilities.

Thus, within its carefully specified and rule-governed context, it was able to generate its own training data and feedback by playing a lot of different games against itself, learning by experience which patterns were mist conducive to success.

## **2.6 The Dwan of Artificial General Intelligence?**

2022 gave us the more significant and widespread shock of AI, with Open Chatbots such as ChatGPT. These ones have been developed using techniques of deep learning, but this time with colossal human textual input in the form of around 300 billion words from various sources on the internet.

And unexpectedly, we are not facing a Technology capable of overcoming the Alan Turing’s’ tests in its full generality, to converse plausibility, flexibly, coherently and informatively about a vast range of topics, and without relying on pre- prepared outputs.

How to assess ChatGPT in theoretical Terms. The systems were developed by applying statistical analysis on those 300 billion words of textual data to create a large language model (LLM) that records the probability that any given sequence of words, will be continued in different ways.

Accordingly, ChatGPT’s primary method of working is to predict which individual words are most likely to follow any particular textual context, and then to choose one of these words. It’s not always the same word since an element of randomness allows to respond differently every time prompted). That word is then added to the text, and the process repeats itself.

Alongside the purely automated learning that generated the trillion or so parameters (i.e. stored probabilities) in the LLM ( reflected in the deep’s network weights, as explained earlier), human users improved the system responses through supervised fine tuning, in which it was given a wider range of typical prompts, together with a suitable human-crafted responses.

Then, a stage a stage of reinforcement learning from human feedback was applied, whereby the system generated a range of responses to each given prompt, and humans assessed the relative suitability of such responses (like or not).

This feedback was then statistically analyzed to generate a “reward model”, which could it turn be used to rate responses in general, and thus assist in selecting the most appropriate. This type of working method is based on imitation with variation/randomness, through the sort of response that it found in the massive textual resources that was used to train it.

The second point regarding this line of working, is that the information it stores implicitly within its trillion or so network weights has been tuned to reflect the characteristics of the textual data, rather than the characteristics of whatever domain the text might concern.

Despite not being able to play games, or any other of difficult game/action because has no mechanisms analysis, lookahead, it is a good representation of general intelligence, about a vast range of topics where it has not been augmented with specific assistance.

Unless the language model in some domain can generate indirectly a reliable model of the reality (which looks more plausible in domains that are primarily conceptual or expressive), It will have nothing like such internal understanding, and can quite appropriately be described as a stochastic parrot.

## **3.0 Machine Learning and its Risks**

AI rose to prominence, in the beginning as a relatively conventional branch of computer science (with well understood data structures and algorithmic methods), but then took a radically different trajectory with the dramatic, accelerating, and previously unexpected rise of contemporary machine learning over the last quarter-century.

Understanding the contrast, is key and beneficial to appreciate why AI is now seen as posing quite distinctive and novel risks in a way that previous computer systems did not.

## **3.1 Four Types of Machine Learning**

**1st is Reinforcement Learning**, in which the machine through trial and error , positive and negative feedback learns how to assess the target variable.

**2nd is Supervised Learning**, in which the aim is to learn from a set of labelled example (Training data) either how to categorize further examples of the same kind of things, or to predict some characteristic. This is long part of the repertoire of classical computation, in the forms of methods such as linear regression, decision trees, and support vector classification.

The problem we are facing today, is that the kind of supervise learning, that is facing a lot of risks, is derived from Deep Learning, particularly because of its inscrutability and hidden basis.

This is ubiquitous in applications to recognize handwriting, faces, and road signs, to analysis of medical scans and visual scenes.

Learned predictions (often called regressions when the relationships involved are straightforward) operates differently, in that now the “labels” are typically numerical values rather than categories, and the aim of the learning process is to be able to predict the corresponding value for new instances (forecasting). (Credit scoring, weather predictions)

**3rd Unsupervised Learning**, where the aim is to identify patterns within the given data, but without the help of human input in the form of specified labels. The two most prominent forms of such learning focus on Cluster analysis and Dimensionality Reduction.

Cluster analysis is a learning technique in which groups of similar objects are automatically identified without needing a training set.

Dimensionality reduction, on the other hand, is a technique in which the main “dimensions” of variation among the objects are identified, enabling their range and relationships to be more easily grasped.

This yet again a form of learning that can be done using classical methods as well as using deep networks.

Principal Component analysis is a notable example of both Cluster analysis and Dimensionality reduction.

**4th is Semi-Supervised Learning,** which is a combination of supervised and semi-supervised learning. It is used in cases where only part of the available dataset is labeled. The unlabeled data is used to determine patterns within the explanatory variables, or it is fitted with “pseudo-labels” determined by estimating a model on the labeled part of the data.

## **3.2 Examples of Unsupervised Learning (Principal Component Analysis)**

Fist example is regarding a physical example. Suppose that we had to plot the position of houses in a village, starting from coordinates in terms of latitude, longitude and height.

PCA will then transform those variables into a different framework, so the various dimensions, instead of lining up with latitude, longitude and height, will instead line up with whatever directions best discriminate between the data items.

Thus, if the village has grown up on a long fairly straight road on mainly flat land, then the dimension that gives most discriminating information, i.e. along which the houses are most spaced out, will be horizontal and in the direction parallel to the road, our first vector.

To provide a proper coordinate frame, the second vector must be orthogonal, i.e. at right-angle to the first, and to preserve most information, we choose another horizontal vector (because houses will vary more in Distance than in height).

Our third vector must be orthogonal to the others and hence vertical.

Other, easier example is by assessing a bunch of authors from century 18 and count how many times they use the most common 50 words, around 16 different texts.

This yields 50 fractional values for each of the 16 texts, representing the relative frequency of each word within it, i.e., if a word occurs 61 times on a 10.000 words text, then the text value is 0.0061. We then imagine these 16 texts placed within a 50-dimensional space, one equally for each word, in which the 50 coordinates for each text are proportional to these relative frequencies.

Thus, each of the 16 texts is represented by a point in space, in such a way that two texts whose points are close together have broadly similar patterns of word occurrence. This allows to see the whichever direction that provides more information, since this whichever direction shows the texts maximally spaced out in a two-dimensional projection.

What is striking about this method is that jt clusters the authors in such a way as to identify patterns of similarities between them, even though it has been done entirely automatically.

It is really unclear to how to understand what similarities and differences are, since there is no simple explaining for Vector 1 and Vector 2 significance, since they do not represent simple words, but hence complicated linear functions of 50 different word frequencies.

To sum up, the PCA provides a powerful method of dimensionality reduction, whereby a large array of points in multidimensional space can be reduced to a graph in two dimensions, while effectively representing the most significant measures of closeness and distance within that multidimensional space.

Despite being hard to pin down the PCA vectors, does not reach the extreme level of inscrutability provided by deep learning networks.

## **3.3 Risk of Inscrutability**

1st when classification or prediction is based on supervised learning with deep networks, it is likely that any bias that exists in the labelling of the training data will be implicitly learned by the model and perpetuated, but in such way that its presence is hidden and hard to eradicate.

For instance, considering a model that allows to select candidates, in a company where the majority of the employees are males, even by not asking directly the gender, the model will use other proxies and would equate into a biased result.

Another related risk is to privacy, because this deep implicit entanglement of complex information within the learned network can easily hold clues to personal characteristics that we would prefer to keep secret.

This also leads to risk of manipulation, in both political and commercial sphere.

All these problems arising from the inscrutability of deep networks models are rendered more intractable because their complex incomprehensibility also makes it very hard to identify, with a view to eradication, the source of any such hidden clues or biases.

Unlike a traditional expert system, the deep network model can provide no explanation of how it reaches its decisions and predictions.

The true explanation is hidden behind the vast web of weights, possibly billions in number, which cannot possibly be rendered humanly comprehension.

Companies that employ deep learning systems without addressing these issues carry huge reputational risk when things go wrong.

## **3.4 Risk of Over Reliance**

Inscrutability can also engender over-reliance, as we come to depend on a system whose mode of operation seems completely mysterious to us, and yet appears to be impressively expert at what it does. This can undermine our autonomy as responsible individuals and lead to serious dangers as we neglect to develop or apply our own judgement and fail to realize when the system is getting things wrong.

Neural networks have long been considered more robust than classical AI alternatives in the context of noisy, ambiguous, or incomplete data.

But to the contrary, it has recently become clear that deep learning models are commonly vulnerable to deliberately designed “adversarial” examples, whereby two images that differ imperceptibly may be categorized by the system as quite different, and even very confidently.

In such type of models, if the task/data involves any nuances, that are less common or entirely novel, then there is a serious risk that the user will end up with a code that sounds quite plausible, but that is incorrect, and it is hard to identify why.

The use of generative AI to generate code doesn’t necessarily guarantee quality code or time savings and could expose a firm to financial risk if managed poorly.

## **3.5 Risk To individuals, Organizations and Society**

Risk poses to individuals vary widely and may originate with organizations or institutions with whom they interact or through their own behavior. If AI systems contributing to decision making are poorly designed or trained, using inappropriate data, then one could suffer because of faulty, biased, or unfair decision making.

Due to the form of who individuals interact with technology, and the tendency of humans to over-rely on automated systems, can lead to complacency and reduced vigilance, which can potentially impact an individual in the form of reduced decision-making autonomy or a risk to safety and well-being.

AI that uses a lot of individual data can be used to manipulate. Risk poses to organizations originates due to over reliance in AI, false sense of confidence, bad decision making, costumer dissatisfaction, and commercial loss. Depending on the type of over reliance, the company may face reputational, regulatory and legal risks.

Advancements in AI create the potential for risks at the society level, mainly job losses, a widening wealth gap between those who have the skills to work alongside AI and the exacerbation of global inequality between countries well positioned to take advantage of AI.

AI Can also be used for actors of bad Ill, such as Deep fakes to mislead and misguide the audience.

In the last few years, the discussions around AI have created the fear of “existential” risk to humanity, specifically in relation about superintelligence and the possibility that AI can use autonomous weapons, making decisions against humanity interest and so on.

It is crucial to be noted, that there is currently no consensus regarding the nature or extent of existential risk posed by AI.

Overreliance on AI systems is also called automation bias.

## **Questions and Answers Module 1 from GARP**

1. What are the Four most Basic Forms of Machine Learning ?

Reinforcement Learning, Supervised Learning, Unsupervised Learning and Semi-Supervised Learning.

1. What are some reasons why the inscrutability of Deep Network Machine Learning is Problematic

The **presence of bias** can be hidden and hard to eradicate. The **Risk to** **Privacy**, because the deep implicit entanglement of complex information within the learned network can easily hold clues to Personal characteristics that we would prefer to keep secret. **Risks of Manipulation**, in both commercial and Political Sphere, because those who can identify our personal characteristics and foibles may also be able to play on them through the now familiar phenomenon of targeted Advertising.

1. What is the risk associated with Overreliance on AI Systems?

Overreliance on AI Systems, also called automation **Bias,** can lead to **complacency** and **reduced vigilance,** which can undermine our autonomy as responsible individuals. This can create the potential for a dynamic in which one neglects to develop or apply one’s own judgment and fails to realize when the system is getting things wrong

1. True or False: The Fact that AI-Specific regulations are still emerging results in organizations having little or not AI- Related regulatory risk

False – Although AI-specific regulations are still emerging, businesses practices that rely on AI are still Subject to existing privacy laws and model governance regulations, so in cases where such practices hurt individuals or groups, organizations are potentially subject to regulatory risk and legal risk.

1. In the Context of Reinforcement Learning, why might Lookahead be applied?

Lookahead becomes essential when our aim is to form a multistep plan to achieve some goal, within a context where there is a very large number of possible situations overall, but with a relatively constrained and predictable range of options in any particular situation.

1. Differentiate between Cluster Analysis and Dimension Reduction

Both are Unsupervised Learning Techniques. Cluster Analysis is a Learning technique in which groups of similar objects are automatically identified without needing a training set. Dimensionality reduction, on the other hand, is a technique in which the main “dimensions” of variation among the objects are identified, enabling their range and relationships (e.g. comparative closeness) to be more easily grasped.

# Module 2 – Tools and Techniques

*Learning Objectives*

Differentiate between machine learning and classical econometrics.

Differentiate among unsupervised, supervised, semi-supervised and reinforcement learning Models.

Distinguish between different data types.

Describe how to encode categorical variables.

Describe how to clean data and the benefits of cleaning.

Describe data preparation techniques and their benefits.

Apply transformations to a set of data.

Discuss how principal Component analysis (PCA) is used to reduce the dimensionality of a data set.

Explain the difference between the training, validation and test data sub-samples, and how each is used.

## **Machine Learning**

Machine learning is a set of tools for data analysis and modeling. Is an aspect of Artificial Intelligence. Covers a range of techniques in which the model is trained to recognize patterns in data. Includes prediction and classification. Gained large popularity on the last decade due to the advances on computer science.

Machine learning offers advantages over traditional econometrics methods in such areas as handling big data, handling non-linearity, reducing dimensionality and handling missing data.

**Handling big data**: ranging from clustering algorithms to neural networks, there are ML approaches that can handle very large amounts of data more effectively than traditional econometric methods. These techniques are highly useful for making use of available data that is growing exponentially in volume, as well as an increase in dimensionality with the rise in digitalization of the global economy.

**Handling non-linearity:** There are many non-linear relationships and patterns in data that traditional econometric techniques might miss, which machine learning tools, such as decision trees, random forests and neural networks can help identify and model.

**Reducing dimensionality:** Machine learning tools such as principal component analysis (PCA) and feature selection can be particularly helpful for prediction and classification tasks when the number of variables is large and when the number of variables is greater than the number of observations, a situation in which standard econometric methods tend to have great difficulty.

**Handling missing data:** There are ML techniques such as K nearest neighbors that can be used to handle missing values in large datasets in a flexible way.

Machine Learning by passes the need for a theoretical reason and goes straight for data processing and is a set of techniques that can work the relationship by themselves, being optimized for predictions.

Machine learning is not on inference, but on the ability to produce out-of-sample predictions. Machine Learning tends to deviate from that of classical statistics.

## **1.1.1 ML, Classical Statistics and Econometrics**

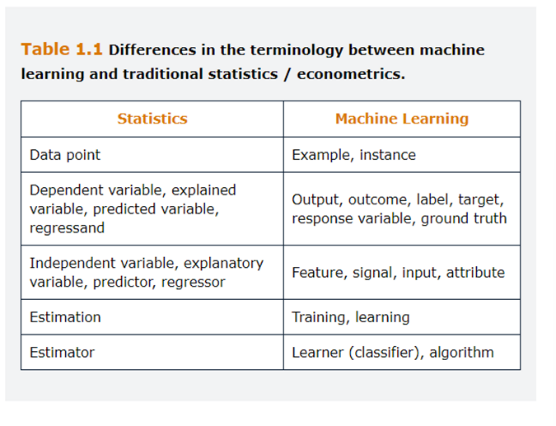
Econometrics is often letdown by bad theorizing at the beginning of the process.

Classical Statistics and Econometrics usually hypothesized that the data generating process can be approximated based on some economic or financial theory. The analyst decides on the model and the variables to include, and the computer algorithm’s role is generally limited to estimating the parameters and testing whether they are significant. Based on the results, the analyst decides whether the data supports the pre-specified theory.

In contrast, Machine learning treats the data generating process as unknown and uses techniques such as regularization to select the relevant predictors.

Model Selection is at the heart of the empirical design of ML applications, and tuning, the process of searching though many models to identify the top performers, is a common characteristic of all ML methods. In contrast, to traditional statistics the focus is not on inference, but on the ability to produce reliable predictions out-of-sample.

Therefore, tools such as measures of out-of-sample predictions accuracy and an understanding of the bias-variance trade off play more important roles than traditional statistics such as R-squared, t-values and p-values



## **1.2 Four Types of Machine Learning**

There are 4 types of machine learning methodologies, reinforcement learning, unsupervised, supervised and semi supervised.

**Unsupervised Learning:** Concerned with recognizing patterns in data. For each observation we have a vector of features, but no corresponding output value to predict. Involves clustering the data or finding a small number of factors that explain the data.

Is not used to generate predictions and at first glance might not appear to be very worthwhile. However, for instances, unsupervised learning could be used by a bank to assess a set of transactions, and check for anomalies that might be suspicious and worthy of further investigation. Dimensionality reduction and PCA are also components of unsupervised learning

**Supervised learning**: Is concerned with prediction and classification. For each observation in the data set, we have a vector of attributes and an associated output or label. The algorithm learns from the “labeled” data with the aim of producing accurate predictions of the target value for new, unseen, and unlabeled instances.

In the financial realm, supervised machine learning found early application, in algorithmic trading and high frequency trade execution. A successful example of classification is in credit decisions.

**Semi-supervised learning**: the objective is to make predictions. But only part of the available data is labeled. The remaining is used to determine pattern within the explanatory variables, or if it is fitted with “Pseudo-labels” determined by estimating a model on the labeled part of the sample.

**Reinforcement learning:** Focus on making a serious of decision to reach a goal. The learning environment might be static, or dynamic, with changes occurring while the process involves. There are no explicit labels. Feedback is provided in the form of reward during the learning process, which encourages a desired behavior, but without giving explicit instructions to the learner. Uses trial and error approach where the desired behavior is rewarded. This is quite useful when decisions need to be made repeatedly so that the algorithm can learn based on the rewards or sanctions received in previous rounds. The output from the reinforcement learning application is a recommended action given the circumstances rather than a prediction, classification or cluster. Is used for instance to find the optimal way to buy or sell large number of shares.

**Parametric vs. Nonparametric:** Machine learning can also be divided into two approaches. **Parametric methods** require the modeler to make an assumption about the functional form of the relationship between the features and the label. This map can be a linear function or a nonlinear, highly complex function. The parameters that describe this map are then estimated or learned using the available data. **Nonparametric methods** do not make any explicit assumption about the functional form of the map or relationship between the features and the output.

**Parametric methods** carry the risk that the chosen functional form is wrong, and therefore the resulting model will not fit the data well. On the contrary, **non-parametric methods** are very flexible and capture very complex data patterns. However, they require many observations to obtain an accurate estimation of the map.

## **1.3 Exploratory Data Analysis**

Is a crucial step before building a machine learning model. Is the process of:

**Collecting Data**

**Cleaning the Data**

**Visualizing the Cleaned Data**

**Analyzing the Final Data**

Data is collected, and then transformed into a usable format and then treated by cleaning duplicates, removing or filling N/A. The Data is then analyzed and visualized to understand any relationship contained in it. Visualization and basic statistical analysis will be useful in selecting features for building a robust ML model.

## **1.3.1 Data Collection and Preparation**

Data analyst often spend 80% of their time cleaning the data, as good data cleaning can make all the difference between successful and unsuccessful ML Project. Data analysis can only be as good as the underlying data, a concept often summarized by the acronym GIGO, which stands for Garbage in, Garbage Out.

**Structured data** are organized in rows and columns, where typically, each column represents one attribute, and the rows contain different *r* observations for the attributes. Census data or the database containing the characteristics of the borrowers of a bank, are examples of structured data.

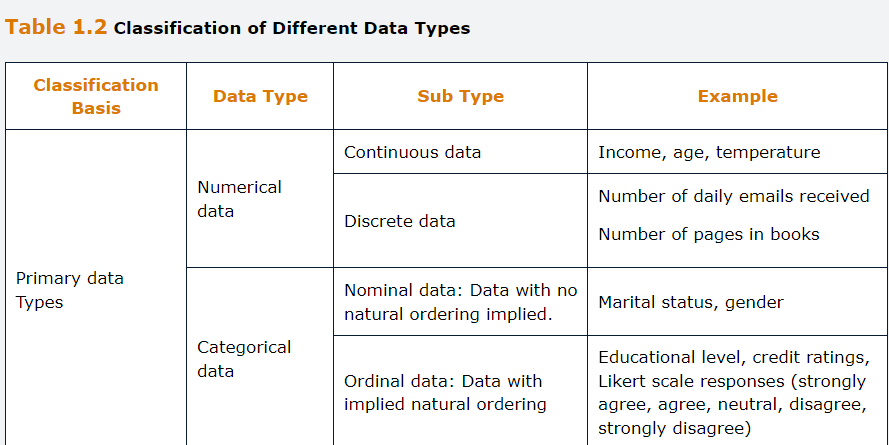
**Unstructured Data** are not arranged according to a preset to a data model. Examples are sensor data, image data, web logs, network traffic, texts… Unstructured data are inherently more difficult to analyze, as they must be represented first in a format that is readable by a machine.

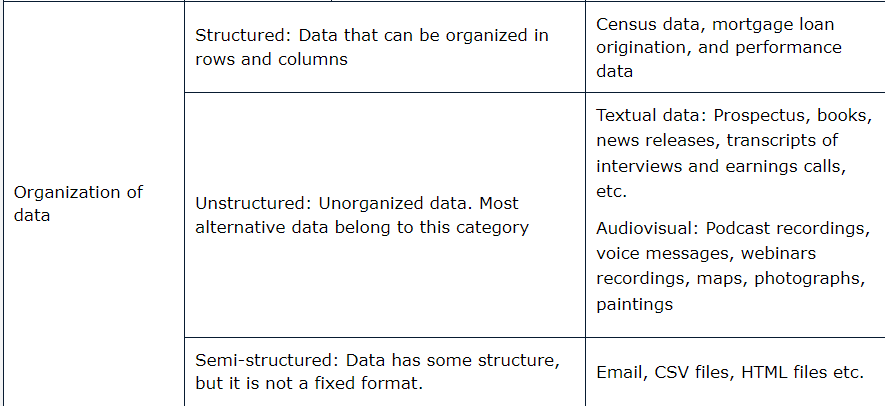
**Semi Structured data** refers to data that is partly structured and partly unstructured, such as geo-satellite images or textual.

**Numerical and Categorical Data:** Structured data sets may contain two types of attributes. Attributes such as age and income are numerical and have a natural ordering, being called continuous numeric or quantitative. Attributes like marital status are said to be categorical and can take discrete values. A special case of categorical variables is binary data. Categorical variables may have or not have a natural ordering.

**Longitudinal variables and cross-sectional variables (data):** The characteristics of a pool of mortgages loans is an example of *cross-sectional data*, where each loan observation is independent of the other observations. On the other hand, *longitudinal data* are related to each other temporarily, spatially and through network connections (time series and so on). Observations cannot be understood independently from one another.

**Textual and Other data:** Natural Language processing (NLP) is a technique that is applied to the process of processing textual data.





A close-up of a computer screen

Description automatically generated

**Nominal Scales:** No order, such as eye color.

**Ordinal Scales:** meaningful order but not having clear interval between variables. Customer satisfaction ratings.

**Interval Data:** meaningful order in whit a consistent interval between values. Does not have a true zero point, implying that a value of zero does not represent absence of the quantity being measured(Temperature, calendars, longitude and latitude).

**Ratio Data:** Implies a natural ordering with a clear interval between values and it has a true zero point. (height and distance traveled).

## **1.3.2 Data Cleaning**

Is crucial due to the errors often originated on the collection process.

**Inconsistent recording:** For data to be read correctly, it is important that all data are recorder in the same way.

**Unwanted observations**: observations not relevant to the task at hand should be removed.

**Duplicate Observation:** These should be removed to avoid biased

**Outliers:** Are observation on a feature that are significantly different form the remaining data, such that suspicion arises that they were generated by a different underlying process. Should be check carefully as they have big impact on the output. Not all predictive models are sensitive to outliers, such as tree-based classification models and support vector machines are deemed to be robust in the presence of outliers. If a model is sensitive to outliers, data scaling can often minimize the problem.

**Missing Data:** Most common problem encountered during the data-preparation stage. Can be structurally missing or just unavailable. Is crucial to understand why the values are missing, and if the pattern of missing data is associated with the outcome. The latest is called informative missingness and can indue significant bias in the model. If missingness is not informative, the removal of a small number of observations with missing data from a large sample is not a problem. Otherwise, one approach is to replace missing observation on a feature with the mean or media pf the observations of the same feature. This technique is called imputation.

## **1.3.3 Data Visualization**

Great to gather patterns and identify potential problems, such as outliers. Are often used to achieve a reasonable understanding of the shape of the distribution of one or more variables. This allows the analyst to detect skewness levels on the data, needs for data transformation.

Figure Below shows a histogram of 521 weekly observations of US Market Returns collected between April 2010 and March 2020.

A graph of a normal distribution

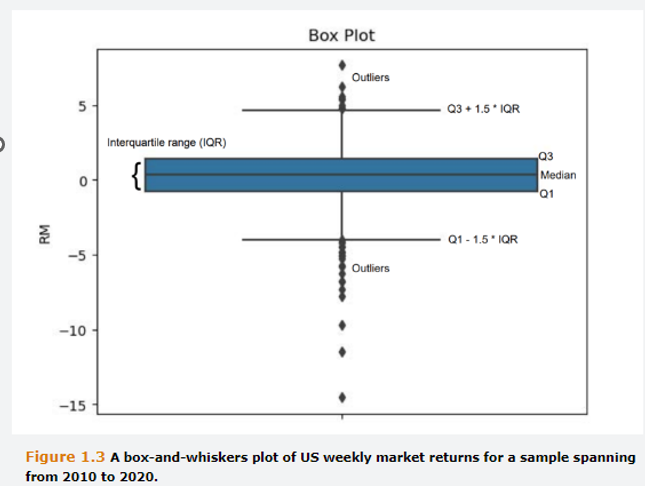
Description automatically generated

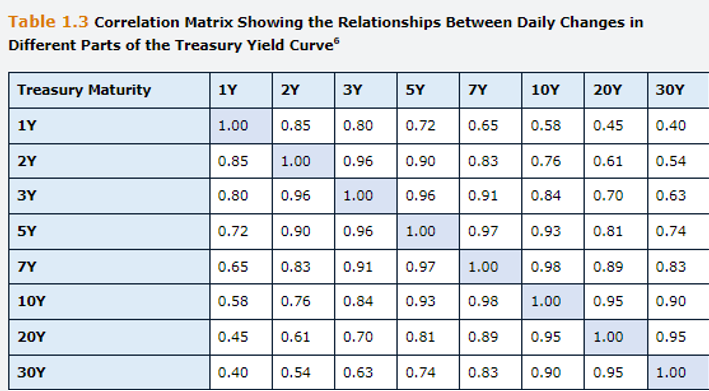
A screenshot of a report

Description automatically generated

**A graph with a red line

Description automatically generated**

****

****

## **1.3.4 Feature Extracting**

Although quantitative data can be directly inputted into a model, qualitative data needs to be transformed in a way that is suitable for statistical analysis.

The process of transforming non-numeric information into numbers is sometimes termed **Encoding**.

Nominal data, attribute a dummy variable in format of 0,1,2,3 and so on. This could equate to problems since one move from 0-1 could be assumed by the model to have the same impact as a move from 1-2 which would probably not be the cases.

In cases such as marital status, (in the case of the marital status example, this could be 0 for “single,” 1 for “married,” 2 for “divorced,” 3 for “other categories,” etc.). Instead of having different categories and dummy’s, we should apply binarization, i.e. for each marital status a 0 or a 1, and the remaining marital status will be accordingly attributed a value between 0 and 1.

As a further example of a categorical variable, suppose we were developing a model to determine whether applications for credit cards should be accepted, and a piece of information we wish to include in the model relates to the applicant’s region of residence in the US. Suppose further that we have five categories: Pacific, Rocky Mountain, Midwest, Northeast, and South. It might be tempting to think that we could set up a single variable taking values such as Pacific = 0; Rocky Mountain = 1; Midwest = 2; Northeast = 3; and so on. However, because the information has no natural ordering, it would be inappropriate to code it as if it did.

Again, the correct approach is to set up a separate 0–1 dummy variable for each category. Then, for each individual applicant, the dummy variables corresponding to the four categories that do not apply would take the value 0, whereas the one that applies would take the value 1.

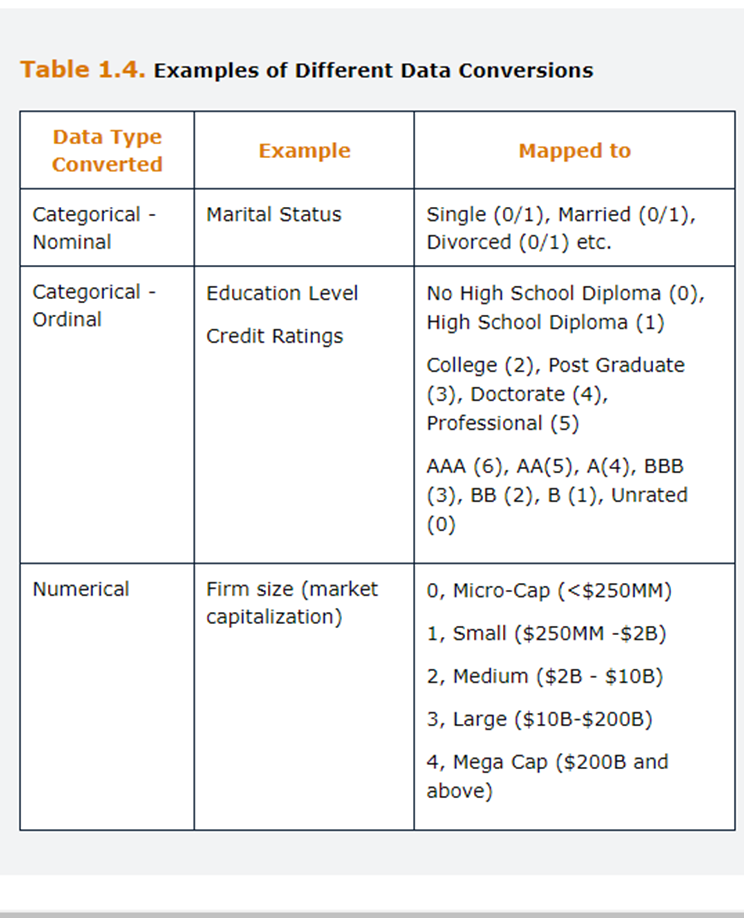
However, including dummy variables for all categories in regression-based models can lead to introduction of **multicollinearity, a phenomenon known as the dummy variable trap.**

This originates when two dummy variables are perfectly correlated, resulting in inaccurate calculations of regression coefficients and standard errors.

In order to solve this, it is necessary to impose constraints on the parameters of regression coefficients. Two commonly used constraints are **omitting one of the Dummy variables from the equation** or **setting constant term (bias) of the equation to zero.**

A slightly different situation is when there is a natural ordering for the categorical data (ordinal variable).

On some occasions, is interesting to convert numeric attributes into categorical ones. **Discretization** is the process of transferring continuous functions, models, variables and equations into discrete counterparties.

****

## **1.3.5 Data Scaling**

Machine learning required that all variables are measured on the same scale. Otherwise, the techniques will not be able to determine the parameters appropriately and the results will be dominated by the feature with the ***largest magnitude.***

**Standardization** involves subtracting the sample mean of each variable from all observations on that variable and dividing it by the standard deviation.

Mathematically, the observation on the variable, , would be changed to:

Where and are the estimated Mean and Standard Deviation, respectively, of the Sample observations on variable i. This process creates a new variable that can take on any value but has a mean of Zero and a variance of one.

**Normalization,** sometimes called the **min-max transformation** takes a slightly different tack, creating a variable that is bounded between zero and one, but that will usually not have a mean of zero or unit variance.

Where and are the minimum and maximum of the observations on variable i.

*Standardization is better when we have Outliers and do not want to exclude them*

The three main reasonings for these processes are:

**Numerical Stability of the learning algorithm**. The difference in the scale of the variables could cause Overflow/underflow easily.

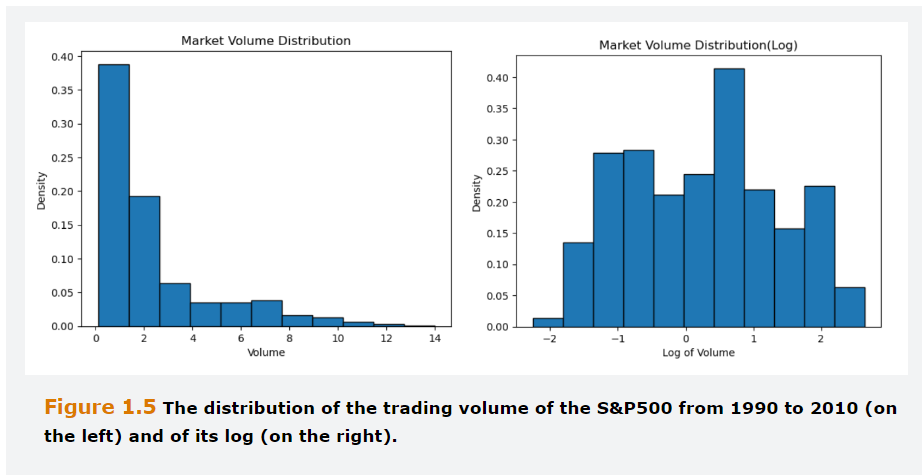
**Ease of interpretation of the model parameter estimations**. If the scales of the variables differ significantly, so do those parameters estimates and it would make evaluation of the importance of each estimate difficult.

**Determining whether the out-of-sample prediction is within the range of the training data**. If the out-of-sample data corresponds to a normalized value greater than 1 or smaller than 0, that means that the prediction is an extrapolation that may correspond to a large prediction error.

## **1.3.6 Data Transformation**

When data is highly skewed it is a good practice to transform it. As a rule of thumb, If the ratio of the highest value to the lowest value is larger than *10*, we consider the variable to be highly skewed. Is common practice to use the Natural Log. Although this not usually produces a symmetric distribution, the date are better behaved. Also, can be applied the squared root of the inverse transformation.

Although scaling does not change the shape of the underlying distributions and correlations, transformation of data will result in changes to the distributions and correlations.

****

## **1.4 Dimensionality Reduction Techniques**

Large datasets can often be represented more compactly which makes the application sophisticated and computationally intensive algorithms easier. Techniques such as PCA use the correlations in the data to represent it in a smaller number of dimensions. These techniques, which also belong to the realm of unsupervised learning, are often used to obtain a smaller number of uncorrelated features, from a larger number of correlated variables.

## **1.4.1 Principal Component Analysis**

Is the most used dimensionality reduction technique. The idea behind this method is to find linear combinations of the original predictors that summarize most of the variability in the data. These linear combinations are called Principal Components. The first PC is the linear combination of the original predictors that captures the most variability in the data among all possible linear combinations.

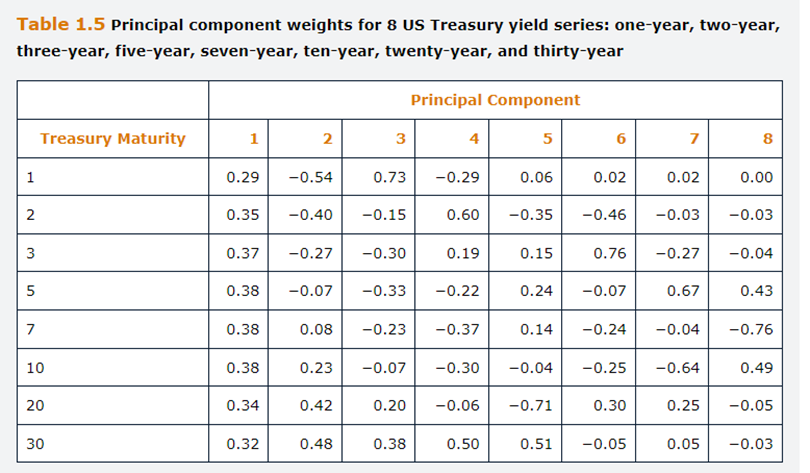
The Second PC is the linear combinations of the predictors that capture the most variability that is not explained on the First PC. This one is constructed to be orthogonal to the first PC, that is, the two are uncorrelated. The process continues the same way to the Third and subsequent PCs, until the variability of the data has been explained, which will provide the same number of PCs as features. Components weights, i.e., the weight given to each PC helps to understand which features are the most important to explain the variability of the data.

The principal Component can be represented as follows:

Where denotates each of the original predictors ( after standardization or transformation), *m* is the number of original predictors and is the weight assigned to in , which captures the importance of predictor *i*  for the Principal Component *j.*

An PCA example is on the yield Curves from US Treasuries with different maturities. From Daily changes ranging from November 2018 to October 2023, using PCA, an analyst can find a small number of uncorrelated variables that describe the Treasury yield Curve.

The Table below shows the Principal Component weights obtained from daily changes for the 8 different maturities. The daily data was scaled before running the principal component analysis. TO explain the movements fully, all eight components are necessary. However, when the actual movements are expressed as a linear combination of the components, the first component explains more than 82% of the variation, and the first 3 components explain more than 98% of the variation. This is due to a high degree of correlation between the yield movements, and the bulk of information contained in them can be captured by a small number of PCs, which are then used as explanatory variables in subsequent regression models, rather than the Yields themselves.



Looking at the Weights above, the first PC loads positively and almost equally on all interest rate variables, which would imply that all the eight yields tend to move on the same direction. So, the first PC can be interpreted as movements on the yield curve level. The second PC Loads negatively, on short term yields and positively on long term yields, implying that as the shorter-term yields rise or fall, the long-term yields tend to move the other way around. Interpreted as movements on the curve slope. The Third PC loads positively on one-year, negatively from two-to ten years and positively on long term yields. We can interpret as a twist in the yield curve.

PCA is extremely useful when the original predictors are highly correlated with each other. In fact, besides reducing the dimensionality of the predictor set, PCA helps solve problems associated with multicollinearity and improves numerical stability of models that require low correlation among the predictors. In fact, PC are built to be uncorrelated with each other and this appealing feature has contribute to their popularity.

It must be understood that PCA seeks to find linear combinations to explain the variability in the predictors without any understanding of the predictor’s measurement scale or their distribution, i.e. if they are skewed. Data should be always scaled, due to PCA attributing more weight to the largest magnitude variables.

PCA is often used in high dimensional problems to choose a lower number of predictors before applying another method. The number of PCs to retain should be provided by the Researcher.

A heuristic way that is often used to support the decision is to construct a plot. The plot usually shows a that the amount of explained variability decreases as the number of components increases (Scree Plot).

## **1.5 Training Validation and Testing**

When a dataset is used for prediction, the analyst wants the model to be able to generalize well to the data that have not yet been used to estimate it.

For this purpose, in conventional econometrics, it is common, although not universal, to retain a part of a data sample for testing the fitted model and determining how well It can predict observations on the dependent variable that it has not seen. This leads to a distinction between in-sample and out of sample parts of the data.

The use of **out of sample (test)** is even more crucial in machine learning, as there is typically little economic of financial intuition behind the modelling assumption and the risk of choosing a complex model that accurately fits the dataset at hand but does not generalize well to unseen data is high.

The estimation of a model that is too complex and captures the noise in the dataset at hand rather than the true nature of the relationship between the features and the output is generally known as **overfitting.**

On the other hand, **underfitting** occurs when significant patterns in data are not captured by the model.

The **Validation set** is used to select between competing models. We are comparing alternative models to determine which one generalizes best to new data-.

Once this model selection has been undertaken, the validation set has already been “contaminated” and is no longer available for genuinely independent test of the model’s performance.

The **test set is used** to determine the retained model effectiveness.

A good model will be able to generalize, which means that it will fit almost as well to the test sample as to the training sample because the machine has learned the crucial elements of the relationship without fitting to unimportant aspects (noise) that would likely repeat in the test set.

## **1.5.1 Sample Splitting and Preparation**

One rule of thumb is that roughly two-thirds of the sample is used for training, and the remaining one third used equally split between validation and test.

If the training sample is too small, this can introduce **biases in the parameter estimation**, whereas if the validation sample is too small, **model evaluation can be inaccurate**, so that is hard to identify the best specification.

If the output data in the sample have no natural ordering **(i.e. they are cross-sectional)** then the three samples should be drawn randomly from the total dataset.

On the other hand, if the data are timeseries, it is common for the training data to be the first part of the sample, then the validation and finally the test. This sample split has the advantage of allowing the model to be tested on the most recent data.

**Cross-validation** involves combining the training and validation data into a single sample, with only the test data held back. Then the combined data are split into equally sized sub-samples, with the estimations being performed repeatedly and one of the sub samples left out each time.

The Technique known as **k-fold cross-validation**, splits the combined training and validation data available, n, into k samples, with the test data excluded from the combined sample.

## **1.6 Software for Machine Learning**

Programming languages are probably among the most important tools in the data analyst’ toolbox.

The best approach for someone who is interested in learning a new programming language is to select one that is flexible with a rich ecosystem of third-party open-source libraries. Such R studio and Python, which are both open-source scripting languages that run on Windows, macOS and Linux platforms and are commonly employed by data analysts for machine learning tasks. Both R and Python can perform virtually every data analyst task, have an easy-to-read syntax and are relatively easy to learn. On the one hand, R tends to be preferred by statisticians as it is great for data visualization and has a rich environment of statistical packages. On the other hand, Python is a general-purpose language, and it is better at non-statistical tasks such as web scraping and textual analysis.

The most important python toolboxes for a data analyst are NumPy, SciPy, Panas, Scikit-learn, TensorFlow and Keras.

NumPy is the building block for many other Python’s toolboxes as it provides the framework to perform operations with multidimensional arrays and to support fundamental linear algebra functions.

SciPy is a collection of numerical algorithms that is used, for instance, to solve optimization problems, a crucial task in machine learning applications.

SciKit-Learn is a machine learning library built on NumPy and SciPy, and it offers tools to perform preprocessing tasks, model training, selection and testing several machine learning methods.

Pandas is a toolbox for data manipulation providing high performance functions for merging, aggregating, and reshaping the data as well as ways to deal with missing values.

A popular package for building and evaluating machine learning models in R is CARET( Classification and regression training). Contains tools for data splitting, preprocessing, feature selection, model tuning and variable importance estimation.

## **Questions and Answers Tools and Techniques – Module 2 GARP**

1. For each of the following terms used in Classical Statistics, provide the equivalent term in Machine Learning parlance:

Intercept = Bias

Slope = Weight

Explanatory Variable = feature

Dependent Variable = Output or Label

In-sample Period = Training Data

Out-of-sample Period = Test Data

1.2

A- What are the Main differences between Machine Learning and more Conventional Econometric Techniques

Under Conventional Econometrics approaches, the researcher selects a particular model or hypothesis and tests whether it is consistent with the available data. The Emphasis is on inference and the main tools are t-statistics, p-values and R-Squares.

Under Machine Learning approaches, the emphasis is on letting the data decide the features to include in the model, with very few assumptions or theory. Inference is less important while the focus is on the model’s prediction or classification accuracy out-of-sample.

B- For what kinds of Problems would machine learning likely be more suitable than conventional econometric modeling?

Machine learning techniques have advantages when applied to problems where there is little theory regarding the nature of a relationship, or which features are relevant.

It is used when the number of data points and the number of features are large (big data or wide data, as opposed to tall data where the number of predictors is strictly smaller than the number of observations).

Machine Learning might also be preferable when the relationship between features are nonlinear.

1.3 What are the main differences between Supervised and Unsupervised Machine Learning methods?

In Unsupervised learning problems, for each observation we have a vector of features, but no corresponding output value to predict.

On the contrary, in Supervisory Learning problems, a set of labeled examples is provided. In other words, there are instances for which values of the predictors and the outcome variables are available.

The Goal is to predict the Outcome for new, unseen, unlabeled instances.

1.4 An example of

A- A Classification Problem

In a classification problem, the label is of categorical nature. An example is to discriminate among credit applicants who will default and those who will not.

The predictors could be the characteristics of the borrowers while the output is a label indicating whether they have defaulted.

B- A Prediction Problem

A prediction problem concerns the prediction of a numerical value. An example is the prediction of the sale of the price of a house.

The predictors could be characteristics of the house while the target variable is the sale price.

* 1. What is an Outlier and How should be treated?

Outliers are observations on a feature that are significantly different from the remaining data, such as being several standard deviations from the mean, such that suspicious arise that they were generated by a different mechanism.

Their treatment depends on the problem at hand. Sometimes they are the fruit of errors in collecting or transcribing the data.

Other times they convey useful information about the data, such as in Fraud Detection.

Remedies include using algorithms that are robust to the presence of outliers or data Transformation.

* 1. What are the benefits of using PCA?

Principal Component Analysis involves projecting a feature dataset onto a smaller number of components. For instance, if the dataset involves ten features, the first five components might be used, which would then reduce the number of input variables by half.

This is particularly useful in situations where the features are highly correlated or very numerous and estimating a model containing them could be challenging.

By construction, the Principal Components are uncorrelated. The technique is straightforward to implement, no matter how many features or data points there are, because the components are simply linear combinations of the features.

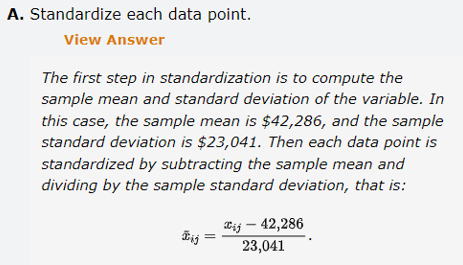
* 1. Standardization and Normalizations.

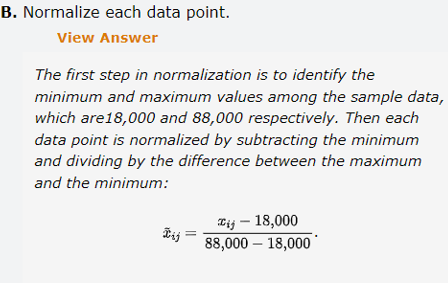
**A screenshot of a screen shot of a number of numbers

Description automatically generated**

**A table with numbers and text

Description automatically generated**

****

****

## **2.0 Unsupervised Learning**

Unsupervised Learning is associated with a model’s use of unlabeled data to develop insights or pattern recognition with no specific guidance or rules.

A common application of unsupervised learning is clustering analysis, also known as segmentation, which aims to separate data points into groups based on the closeness of their features.

Clustering places points that are similar into the same group and points that are dissimilar into different groups. There are many applications of clustering analysis including:

Organizing customer data into groups to determine the characteristics that separate their purchasing behaviors. Investigating whether subsets of banks accounts can be clustered into groups likely and unlikely to involve fraudulent transactions or money laundering, which is an example of anomaly detection.

Creating cluster of documents and newswires with similar content. Cluster is also helpful for identifying the structure of a set of features prior to conducting a classification or prediction task.

In other words, even where we have labelled data, we might choose to deliberately ignore the labels initially to focus first on better understanding the characteristics of the features.

All Clustering applications are based on measurements of distance, as explained below, and consequently it is essential that the data are normalized or standardized before analysis.

Otherwise, if one of the features has a larger scale than the others, it will end up dominating the measures so that other features are rendered irrelevant.

There are different types of clustering. One set of techniques is hierarchical clustering, where we begin with either one cluster containing all points and successively separate them into sub-clusters, or alternatively we begin with each data point In a separate cluster and successively combine them together.

The former approach is known as divisive clustering, while the latter approach is called agglomerative clustering.

A dendrogram is an example of a hierarchical clustering technique. A second set of clustering techniques is partitional clustering which separates the dataset by locating observations based on their centroids, which are center points of the cluster.

The K-Means clustering algorithm is an example of such techniques. A final set of clustering techniques is based on density points in feature space, of which DBSCAN (density-based clustering non-parametric algorithm) and SNN (shared nearest neighbor) are examples of.

Inertia is an example of a general approach to understanding the number of clusters, because the better the model fits, the closer the data points will be, collectively, to their respective centroids.

Calculate the Euclidian distance measure between a data point and the centroid to which it has been allocated. With this, Inertia or Within-Cluster sum of Squares (WCSS) is:

Where is the number of datapoints within the Kth Cluster. The Total WCSS for all the clusters is given by:

The lower the Inertia, the better each cluster fits the data, the more separation between clusters.

The Inertia measures within cluster distances.

The Silhouette Scores compare within cluster distances and the distance between cluster, and the ideal cluster is a set of clusters in which the points within the cluster are close to the centroid and the clusters are quite far apart from one another.

The silhouette scores compares the average distance of each observation from other points in its own cluster (known as cluster cohesion, denoted ai, whit its average distance from points in the closest other cluster, denoted bi.

For each observation will lie within the range [-1,1].

By averaging the silhouette scores for all the data points within a cluster, we can calculate one silhouette score, , for each of the K clusters. It is then the common average of these scores:

A better value of K is the one that gives a higher silhouette score s, which implies that the points within a cluster are closest together but furthest away from other clusters.

Problems for K means are the data and the methods, which assumes that data points should be spherically aligned across a centroid, which is not always the case.

A graph with blue dots and a circle

Description automatically generated with medium confidence

Other big issue is the presence of outliers which can cause distortion of how our data looks like.

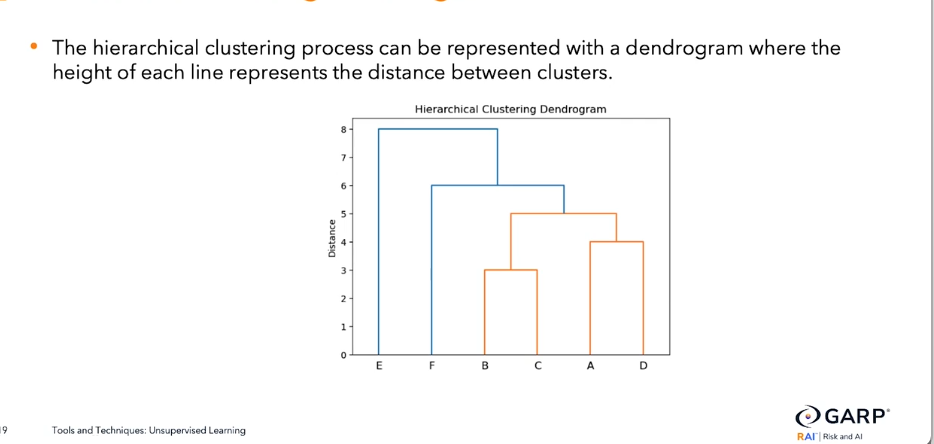
K cluster handles big data quite well but not many features, being a.k.a. the curse of dimensionality.

Hierarchical clustering does not require a priori specification of the number of clusters as K means does. Instead, it utilizes every possible number from 1 to Number of features.

To measure effectiveness instead of distance measure uses a linkage criterion, such as single linkage which calculates the distance between clusters based on the distance between two data points from those clusters that are closest to one another, while complete linkage calculates the distance between data points in the clusters that are the furthest apart from one another.

A diagram of a couple of circles and lines

Description automatically generated



## **2.1 K-Means Clustering Algorithm**

Is a straightforward, unsupervised, algorithm to separate N observations into clusters. The number of required clusters, K, is determined at the outset by the analyst.

Often analyst, try several different Values of K and them aim to choose the most appropriate from among them.

The algorithm sometimes also known as Lloyd’s algorithm proceeds as follows:

**First,** randomly, choose initial values for the centroids, which are the centers of the Cluster.

**Secondly**, allocate each data point, to its nearest centroid.

**Thirdly**, recalculate the centroids to be at the centers of all the data points assigned to them.

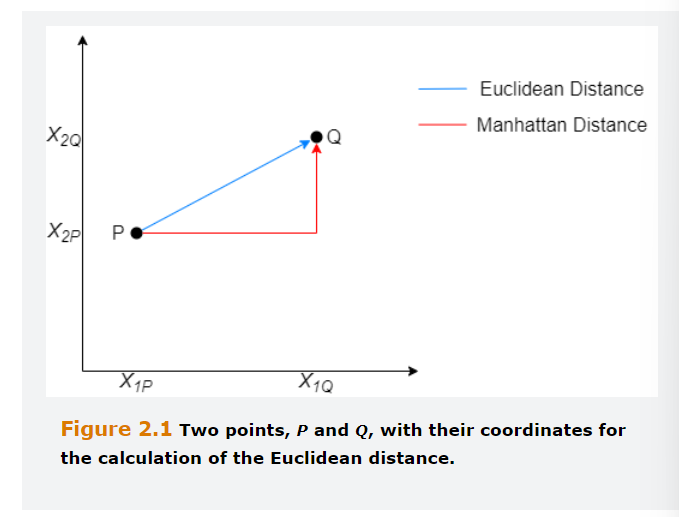
**Fourthly**, repeat steps 2 and 3 until the centroids no longer change.

Instead of centroids, the cluster can be determined with reference to medoids, which are the most frequently occurred points, if the features are categorical rather than taking continuous values.

Steps 2 and 3 require a definition of distance of each observation to the centroids. There are two commonly used measures. The first is the Euclidean (“as the crow flies”) distance, and the second is the Manhattan distance measure.

For illustration example, supposed that we have two features, X1 and X2, and two observations on each of them, represented by the points, P and Q, which have coordinates, The Euclidean distance, , between the two points would be calculated as the square root of the sum of the squares distance in each dimensions.

The measurement would be constructed in the same fashion if there were more than two dimensions. If there were m features for two points P and Q, the distance would be the square root of the sum of the square’s distances



Note that for simplicity the formula above describes the distance between one point P and another point Q. However, the purpose of K- means is not to minimize the distance between points, but rather to minimize the distance between each point and its centroid.

A screenshot of a math test

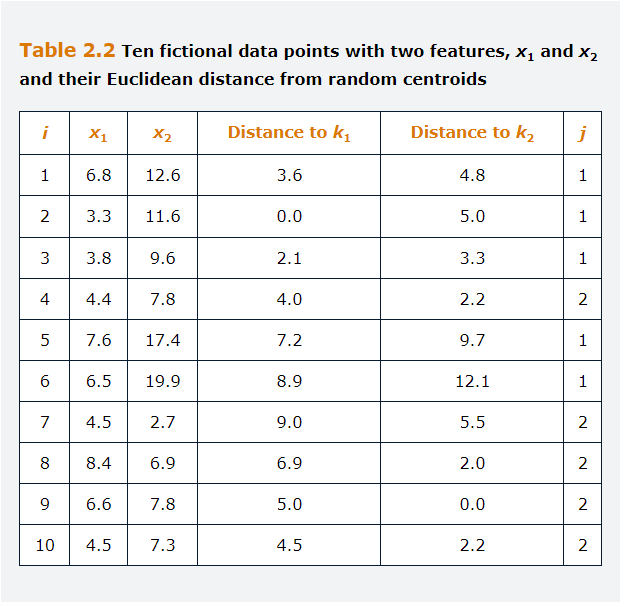
Description automatically generated

Although the K-means algorithm should be applied on data that underwent a rescaling of some from, for the sake of this example we skip that step, given that the two features are measured on a similar scale.

The Euclidean Distance between i=1 and the first centroid (i=2)

The Euclidean Distance between i=1 and the second centroid (i=9)

Table 2.2 reports the Euclidean distance between each datapoint and each of the two centroids. Each point is assigned to the closest centroid.



Obviously, the points from where we started are no longer centroids of these two newly formed clusters. Therefore, we should now compute the new centroids based on our assignment in the first step. The centroids are computed as the averages of the features of the points allocated to that cluster

For the first Cluster the average of X1 is:

For the first Cluster the average of X2 is:

The new centroid of cluster 1 is [5.6,14.2]. For cluster two would be [5.7,6.5]. We can now compute the distances of each data point to the new centroids and assign each observation to the cluster with the closest centroid. The algorithm stops when the clusters no longer change. In this case, in the following interaction in the point [3.8, 9.6] mover from cluster one to cluster two. After this iteration the cluster no longer changes, so the algo stops.

A graph of a function

Description automatically generated with medium confidence

## **2.1.1 Performance Measurement for K-Means**

The first option is to rely on a measurement called inertia, i.e. WCSS, in which the better the model fit’s the closer the data points will be, collectively, to their respective centroid.

The Between Cluster Sum of Squares (BCSS) which is the sum of squares of the distances of each centroid to the centroid of all data points weighted by the number of data points in each cluster. This measure increases as the number of Clusters grow.

A slightly more sophisticated approach to performance measurement is the to use the Variance Ratio Criterion (VRC), which is a.k.a the Calinski-Harabasz index:

Where N is the total number of data points. A higher VRC implies a model with better grouping of clusters.

## **2.1.2 Selecting the Starting Positions of the Centroids**

Choose Randomly. If we choose a different set of initial allocations, it is likely that we get a different set of solutions, particularly for large values of K.

To mitigate the impact of the initial selection, is common practice to run the K-means several times with different random initialization, and then select the one with lowest inertia.

A further alternative is to use K-means ++ which involves establishing the initial position of the centroids far from each other in the feature space, so that the position of only one of the centroids is choose randomly. This can lead to better outcomes and faster convergence to the optimal solution.

Actually, randomly selected Centroids could mean that points are close to each other and make it harder to distinct between Cluster.

## **2.1.3 Selection of K**

In the same way that R squared will never fall when more explanatory variables are added to a regression model, the inertia will never rise as the number of centroids increase.

In the limit, as K tends to N, each data point will have its own cluster and the inertia, WCSS, will fall to zero.

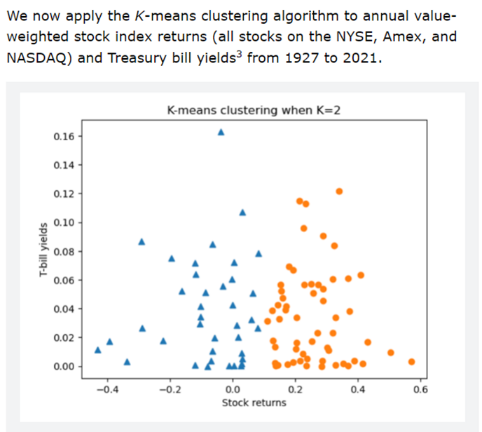
A straightforward rule of thumb is to set K equal to the integer closets to the square root of half the number of data points,

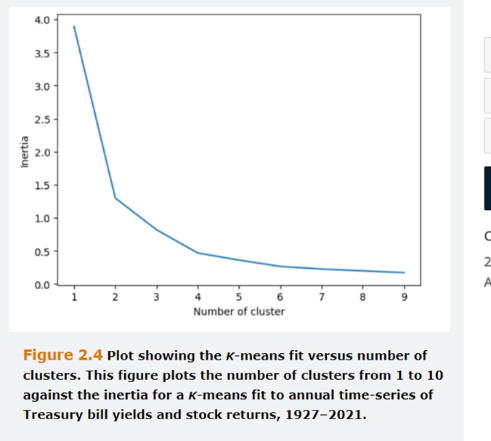
However, this is rather arbitrary and will not vary depending on the characteristics of the data set.

There are two better alternatives such as Scree Plots and the Silhouette method.

On Silhouette score, which ranges from [-1;1], if the score is equal to 1 would imply that all the points allocated to each cluster were exactly on the centroid of their respective cluster, and s=0 implies that the cluster are significantly overlapping. S<0 is empirically unlikely but would correspond to points being mis-clustered.

## **2.1.4 Selection of K Example**





## **2.1.5 Advantages of K Means**

Is widely applied in finance and many other areas. Compared with other clustering techniques, its advantages include:

**Firstly,** makes Intuitive sense and is easy to visualize if there are up to three features.

**Secondly,** is quite fast and scales well to large number of data points

**Thirdly,** will always converge to a solution, even though it might not always be the most appropriate solution

## **2.1.6 Problems with K Means**

**Non-spherical Clusters:** Since it is based on distance from the centroid, tends to form spherical clusters. 1st problem is if a cluster is within other cluster, because the algorithm will be unable to differentiate, the 2nd problem is that sometimes points within a cluster are closer to the other centroid than from the one from its own cluster.

A solution for this is to either apply a non-linear transformation to the input data, known as Kernel Function, or to switch to a different technique that can cope better with a wider variety of data patterns.

**Presence of Outliers:** Outliers might generate their own cluster or create distortion on the cluster centroids, especially if K is a high value. Standardizing the data using min-max will mitigate this issue to some extent, although from this perspective it may be preferable to remove outlying data points entirely from the sample prior to beginning the analysis.

**Curse of Dimensionality:** Tends to perform poorly as the number of features increases. In order to solve this problem, some features can be removed, the features can be transformed to a smaller subset using Principal Component analysis or similar, employ the cosine distance, which is a measure that does not always increase with respect to the number of features that can be employed instead of the Euclidean Distance.

## **2.1.7 Fuzzy K Means**

All the above involves hard clustering, where data points are allocated uniquely to each centroid, meaning that a particular data point is either in cluster j or it is not. Each point is allocated to one and only one cluster, which is known as Winner takes all clustering regime.

An alternative is soft clustering, also known as as Fuzzy clustering or Fuzzy C means, in which data point can belong to one or more clusters, with a probability assigned to each cluster.

By applying this is necessary to change the inertia equation, because of calculating the direct summation of , the equation incorporates a probability raised to a power f, in which f is the fuzziness Coefficient. The large F the greater the extent to which the clusters will overlap., i.e., they will be fuzzier.

## **2.2 Hierarchical Clustering**

The limitation of K-means clustering is that necessitates the number of clusters being specified a priori.

Hierarchical clustering does not require this, instead utilizes every possible value of K (from 1 to N) as an Integral part of the process.

**Divisive:** This begins with just one cluster and them sequentially partitions the data by adding another cluster until every data point has its own cluster.

**Agglomerative:** This begins with each point having its own cluster, and then the Closest two cluster in feature space are merged. The process ends when all points have been merged into a single cluster.

In these two processes, we will need to capture the distance between one cluster and a single point on that cluster.

Instead of a distance measurement, a linkage criterion is used, of which there are several variants. The most straightforward of these is a single linkage, which simply calculates the distance between clusters based on the distance between two data points from those clusters that are closest to one another.

Another possibility is to use a complete linkage, which calculates the distance between the points in the cluster that are furthest apart from one another.

A diagram of a cluster

Description automatically generated

When the process of division or agglomeration is completed, the output is presented in a dendrogram, which has the data points on the X axis and distances on the Y axis.

A diagram of a clustering diagram

Description automatically generated

The example above is related to the stocks and bond example from the K-means topic.

If we specify a cutoff so that only three levels from the original split are shown to avoid proliferation of tiny clusters culminating in data points at the bottom.

The value in parentheses shows the total number of points that are within a cluster, whereas the value without parentheses corresponds to the individual data for a specific year.

The distance is calculated using a slightly more sophisticated method known as Ward linkage, which minimizes the variance of the clusters being grouped.

To select the optimal number of clusters, we examine the heights of each vertical line before a split occurs, which shows how much the distance drops as an additional cluster is introduced.

In this case, if splitting the data into two clusters provides a large drop in distance relative to others, it would suggest that a division of cluster is preferred. But if the line is small, would suggest the incremental benefit in terms of reduced distances is not worthwhile. Like a Scree Plot, a dendrogram will not be able to show a uniquely correct optimal number of clusters, rather the optimal number will be a matter of interpretation.

## **2.3 Density Based Clustering**

Another approach to clustering is the family of techniques including DBSCAN (Density based Spatial Clustering of applications with noise) and SNN (Share Nearest neighbors) that are based on density points. Here, a region in feature space constitutes a cluster if the density of points exceeds a pre specified threshold. DBSCAN distinguishes between three types of data points:

**First,** an Observation is a core point if at least a pre-specified number of other points are within a threshold distance of it.

**Second,** an observation is a border point if it is within the threshold distance from a core point, but it has fewer than the pre-specified number of other points close to it.

**Third,** an observation is considered a noise point if it is neither a core nor a border point.

Each core point constitutes a cluster, and the border points are allocated to their nearest core point cluster. Noise points are ignored and remain un-clustered. The threshold and number of points required to assign an observation as core are two hyperparameters to be tuned.

Although more complex – in terms of both intuition and the optimization method – density-based clustering has two important advantages. First, it can handle non spherical distributions of points in feature space. Second, it is considerably more robust with respect to outliers, and indeed observations identified as noise points are automatically excluded from the clusters.

## **Extra 2.A Different Distance Measurement**

**Manhattan distance**, also known as L-Norm, between Point P and Q

Extending to m dimensions:

The **Euclidean Distance** is the Direct Route, whereas the Manhattan measure gives an approximation to the distance between two buildings that might be required when driving a car.

**Minkowski Distance** consists of nesting both the Euclidean and Manhattan distances in a broader framework.

If L=1 the Manhattan measure is calculated, if L=2, the Euclidean measure and if finally, L=, it is calculated the **Chebyshev Distance**, this is the maximum of the absolute distances along each dimension.

**Cosine Similarity,** is the cosine of angle between two vectors, P and Q

Expressed in Scalar form

Which is bounded to lie in the [-1;1] interval with all notations as above. In machine learning this measure is used to analyze the similarities between documents.

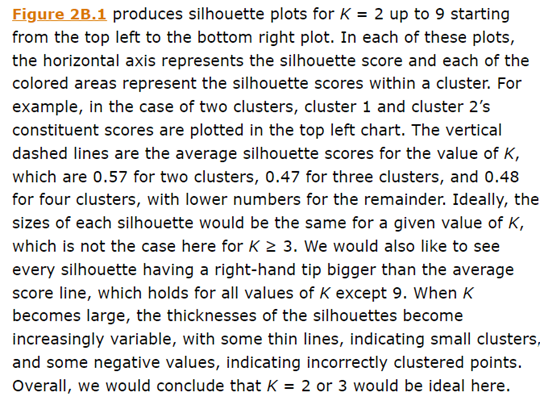
The **Cosine Distance** is defined as the complement of

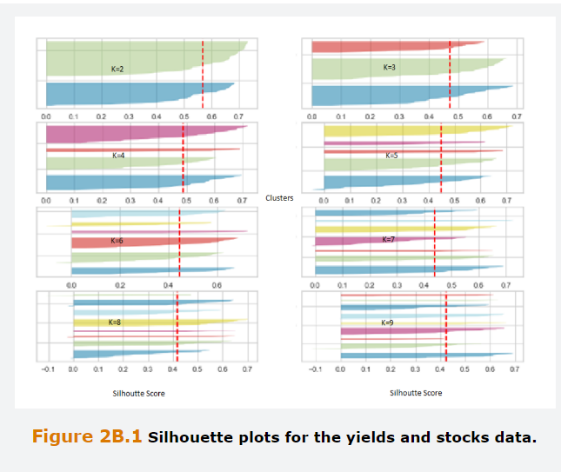
Which is bounded to lie in the [0,2] interval.

**Mahalanobis Distance** is the distance between a data point and a distribution. Is the multivariate equivalent of the Euclidean Distance. It can be also viewed as a multivariate version of the Z-Score.

Where x is the vector of observations, miu is the vector of means, and Epsilon is the variance covariance matrix. The distance measure is used for cluster analysis and classification as well for detecting outliers.

## **Extra 2.B Silhouette Method**





## **Extra 2.C K-Means Clustering Example**

A retail bank is interested in examining the Characteristics of its customers who take a loan to buy a car. It identifies two features, client age and loan amount in US Dollars, for six costumers. Data on the Table below is raw and Standardized.

Use this data to form two clusters using the K-means algorithm, integrating twice with initial centroid positions, assuming they were randomly selected, of (0.5,0.5) and (1,1) in standardized feature space.

A table with numbers and text

Description automatically generated

The first stage is to work out the distances from each point individually to each of the centroids. Then a point is allocated to the Centroid to which it is closest. We work with the Standardized features and assume Euclidean Distance measure is used.

The calculations are straightforward (Just use the Euclidean distance for each data Point).

Once calculated the distances, attribute this point to the respective Cluster.

A screenshot of a graph

Description automatically generated

Having Completed the allocations to first stage Cluster, we then reposition the Centroids at the Centers of those clusters by computing the average of each co-ordinate over the data points allocated to that cluster.

Because there is only one data point allocated to that cluster (1,1)the position of the centroid will shift to the coordinates of that data point, namely (1.25,1.34).

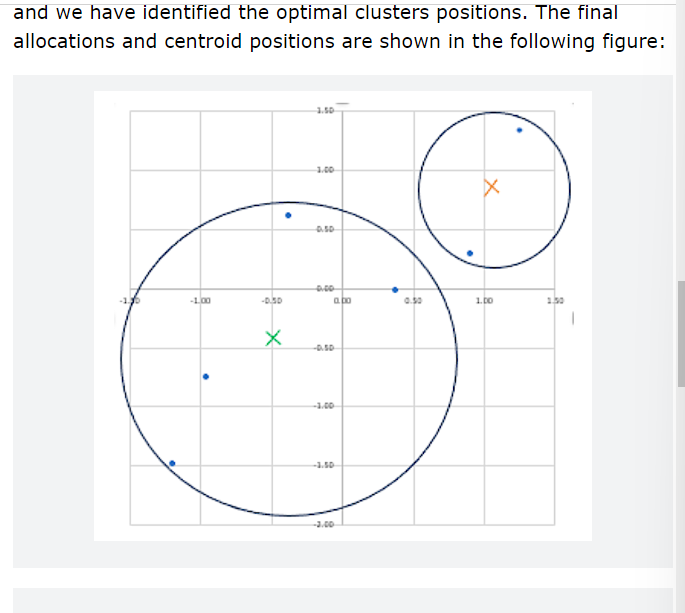
We take an average of each of the features across the other five data points to get (-0.25,-0.27)

Next, we repeat the process by determining which of the two new centroids each point is closest to.

Point 4 now switches to the other Cluster, but the other four points are still allocated to the same cluster as in the previous stage.

We then recalculate the positions of the centroids, which will differ due the reallocation of Point 4.

It turns out that when we calculate the distance of the points to each of these new centroids, none of them switch clusters and therefore that is the end of the process, and we have identified the optimal clusters position.



A table with numbers and letters

Description automatically generated

A table with numbers and a few words

Description automatically generated with medium confidence

A table with numbers and text

Description automatically generated

## **Questions and Answers Module 2 Chapter 2 from GARP**

***2.1*** *How would you choose the number of Clusters when using unsupervised Learning?*

The more clusters are used when fitting an unsupervised Learning model, the better the fit of the algorithm to the data, but as the number of clusters increases, the usefulness of the models starts do diminish.

Determining the most appropriate number of clusters for a particular dataset could involve constructing a “scree plot”, which charts the Inertia (Sum squared distances of each point to its centroid), against the number of clusters. We would then search for the number of clusters beyond which the inertia only declines very slowly. Silhouette Scores, which compare the distance of each point (a) to points in its own cluster and (b) to points in the Closest other Cluster, can also be used.

***2.2***

***A- Explain the steps in using the K-means clustering algorithm***

First, specify the number of centroids, K and choose a distance Measure, such as the Euclidean or Manhattan Distance.

Secondly, scale the features using either standardization or normalization.

Thirdly, select K points at random from the Training data to be the centroids.

Fourthly, allocate, each data point to its nearest centroid.

Fifthly, Given the points allocated to each centroid, recalculate the appropriate location of the Centroids.

Sixthly, if the positions of the centroids have changed from those in the previous iteration, then repeat step 4. If the positions of the centroids have not change (and the clusters are not changed) then stop.

***B- In practice the K-means clustering algorithm is often carried out with several different initial values for the centroids. How would you choose between clusters that result from different initial choices of centroids?***

You could select the centroids where the total inertia was the lowest, as this would represent the choice of centroid positions that best fitted the feature data.

***C- How do you use the Scree Plot for choosing the number of K-means Cluster.***

A Scree Plot allows you to check the “Elbow” in the plot, where its gradient changes from steep to almost flat and that is the optimal point of K.

***2.3 What are the two types of hierarchical clustering? What are the advantages of hierarchical Clustering?***

Hierarchical Clustering starts with all points in one cluster and then sequentially splits them into separate clusters until an optimal allocation is reached (divisive Hierarchical Clustering) or starts with each data point in its own cluster and sequentially combines them until an optimal allocation is reached (agglomerative hierarchical clustering).

The Advantages of hierarchical clustering are firstly, that it does not require a pre-defining number of clusters and secondly, it can uncover hierarchical relationships within the data, which can reveal nested clusters within larger groups. Thirdly, the dendrogram produced are really straightforward to interpret.

***2.4 State if True or false***

***A- K-means with Euclidean distance can only be used when the clusters are approximately spherical.***

True, because K-means is based on Linear Euclidean distances, it runs into problems when the cluster is not approximately spherically shaped. When the Manhattan distance measure is used with K-means, the clusters are approximately rhombus-shaped (losango). This can result in poorly defined clusters, or some points even being allocated to the wrong clusters.

***B- The WCSS will never rise when the number of clusters is increased in a K-means application***

True, the situation is analogous to what happens to the residual sum of squares when more features are added to a linear regression. When additional clusters are added (i.e., the value of K increases) the fit of the model to the data cannot get worse. Therefore, the WCSS must fall as the new cluster will capture one or more o the data points better than the cluster to which it was previously allocated.

***2.5 What is a Dendrogram and how would we interpret one?***

A dendrogram is a pictorial representation of the steps in a hierarchical clustering application, which shows how the clusters are split or combined. Each bifurcation (for divisive clustering) or combination (for agglomerative clustering) of the lines shows a cluster being formed or removed, respectively. The heights of the vertical lines show the impact of the marginal cluster on the distance of the points affected by it to their nearest cluster. If a particular vertical line is long, this would suggest that the additional cluster has a considerable effect on the model fit and therefor is worth incorporating.

***2.6 Determine the Centroid of a cluster comprising Banks A,B and C using the Raw (unscaled) data.***

A table with numbers and words

Description automatically generated

The centroid is simply the average of the feature values across the three banks. So, if we define the coordinates space as three dimensional point, with the raw data in their original units, the centroid is given by

Which is (2.567, 12.333, 176.667)

## **3.0 Supervised Learning for Numerical Data**

Learning objectives:

Identify uses and limitations of single and multi-variable linear and non-linear regression.

Interpret the results of single and multi-variable regression and non-linear regression.

Identify problems that may occur with linear regression models and possible remedies for them.

Describe how logistic regression models can be applied to classification problems.

Describe the use of linear discriminant analysis for classification problems.

This chapter covers models that used for supervised learning arising from econometrics, techniques originating from computer science and more commonly associated with machine learning.

***Single Variable and Multiple Variable Linear Regression***

The simplest linear regression model is one that has only a single feature or input, variable x along with the output or target variable y.

Using multiple variable linear regression, it is possible to explore interactions between variables by incorporating interaction terms. It is also possible to incorporate power terms in a multiple variable regression.

The Main methods for estimating the parameters of the regression model are:

**Least squares**

**Maximum Likelihood**

**The method of moments**

A specific case of the first technique, least squares, it known as the Ordinary Least Squares (OLS), which is the most straightforward and commonly used approach for linear regression models.

Simple Linear regression is given by:

While Multiple linear regression

Ordinary least squares, works to estimate the correct linear relation as trying to minimize the error of using x to predict y.

***Problems with features and functional form***

The model omits relevant features such as a lack of Data or lack of awareness of their relevance, and the parameters being estimated

The model includes irrelevant features, then estimate precisely and is hard for the model to generalize from the specific training sample to the test sample.

The model incorporates features in the wrong way, known as incorrect functional form,

***Multicollinearity***

Perfect Multicollinearity is when two features have an exactly linear relationship such as , x2=10x3

Near Multicollinearity is when two or more of the variables are closely, but not perfectly, correlated with one and another.

Techniques for addressing near multicollinearity include:

Removal of one or more highly correlated variables

Turning the highly correlated variables into a ratio or difference

Use regularization

***Outliers***

There is no widely accepted formal definition of an Outlier, and in broad terms refers to an anomalous data point that lies a long way from the others. They can have a considerable effect on the estimated parameters.

In order to detect outliers we can use a Residual Plot, in which we examine the plot of the residuals (the difference between the actual data points and the corresponding values fitted from the regression line), and noting any point that lie further from the line than others.

We can also use the Cooks Distance, which measures the influence of each individual data point on the parameter estimates. Achieves by removing each data point separately from the regression and determining the difference in model fit for all the remaining points. The bigger the Cooks distance the more influence the data point has on parameter estimation.

***Heteroskedasticity***

Refers to non-constant variance of the error term and can lead to inefficient parameter estimation and errors in the determination of the statistical importance features.

A plot of residuals against fitted values can be helpful in identifying heteroskedasticity.

In a good Model, a model that is Homoscedastic, we will see errors all clustering around zero.

If the model is highly heteroskedastic, it’s probably because we lack using a relevant variable.

***Classification Problems***

There are many instances where a model’s output (dependent variable) is categorical.

Predicting a qualitative outcome is defined to be a classification problem and assigning an observation to one class rather than another is referred to as classifying the observation.

A specific case of categorical data is where the output is binary, that is, it only has two outcomes.

We might be interested in modeling the probability of one of the outcomes occurring.

One outcome (referred to as Positive outcome) is assigned a value of 1 and the other (referred as negative outcome) is assigned a value of 0.

A Standard linear model is inappropriate in such case because there would be nothing in the models design to ensure that the estimated probabilities lie between zero and one, and we could obtain nonsensical predictions.

***Logistic Regression***

A Logistic regression uses a cumulative logistic function transformation, resulting in the output being bounded between zero and one.

The logistic function has a sigmoid shape and is written by

When there are m features, the functional form is estimated as,

**A graph of a function

Description automatically generated**

***Linear Discriminant Analysis***

A Logistic regression works really well for binary classification problems.

When there are multiple, well separated classes, logistic regression estimates can be very unstable. In this case, an alternative is offered by LDA.

Like with logistic regression, the idea is to assign each instance to the class with the highest conditional probability.

A discriminant function is calculated for each of the classes which gives the probability that a new data point belongs to that class.

New data points are classified based on which class has the highest probability.

## **3.1.1. Simple Linear Regression**

Also known as bivariate regression because there are only two variables.

Simple Linear regression is given by:

The model postulates that y varies due to changes in x, here y is a linear function of x and an unobservable error term, u, with mean zero and constant variance.

x and y are observable variables (y is the target and x is the feature, in machine learning parlance), where as are the parameters to be estimated.

Using econometric terminology is the intercept parameter, and it is interpreted as the value that y would take if x equals to zero. is the slope, and measures the impact on y of a unit change of x.

In machine learning, the intercept is known as the bias and the slope is the weight.

There are three main methods for estimating the parameters of a regression model:

Firstly, **Least Squares**

Secondly, the **Maximum Likelihood**

Thirdly, the **Method of Moments**

The Ordinary least Squares is the most straightforward approach and hence it is most used for linear regression models.

A linear regression model embodies a linear relationship that can be represented by a straight line. Being slightly more specific, the model is both linear in the parameters and linear in the variables.

In order to use OLS, the model must be linear in the parameters, although it does not necessarily have to be linear in the features.

## **3.1.2. Multiple Linear Regression**

In the majority of cases, having a model with an unique feature is not sufficiently flexible to capture all the variability in the target variable, and we can build a much better model by considering multi variables.

In the multiple linear regression, there will be m+1 parameters to estimate. One for the intercept, and one for each of the m slope parameters.

Once we have data on y and , again, OLS can be used to estimate the Parameters.

In the multiple linear regression model, each parameter measures the partial effect of the attached variable after controlling for the effects of all the other features included in the regression.

Even within this straightforward framework, we can nonetheless incorporate a wide range of specifications. For instance, it is common to apply the logarithmic transformation to some or all the feature variables and/or the output variable.

Such a transformation would imply a different interpretation of the parameter estimates but OLS could still be used as the model would remain linear in the parameters.

Alternatively, we could incorporate interaction terms, i.e. features multiplied together, or power term of features:

Where, y depends not only on the levels of and , but also on how they work together. Hence, will capture any complementarity between them, where changes in both variables are required to have an impact on y rather than in isolation.

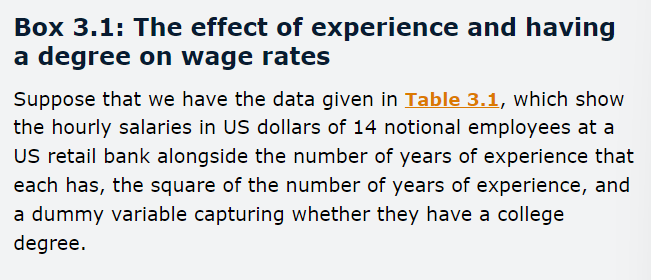
A commonly employed example is how the amounts of water and fertilizer affect crop yields, because and abundance of one and none of the other will not lead to high yields. Hence, the amount of one of them influences the effectiveness of the other, implying a need to model them jointly.

A model including power term could be as follows:

Here, only a squared term on is included on the model, and it is common to stop there, allowing for a quadratic relationship between .

But, in principle, would be feasible to include cubed terms, fourth-order powers, and so on. However, when adding further terms, one must be mindful not to overfit the data.

To use OLS for model estimation, the output variable y must be continuous, but the features could be continuous or discrete, being that the discrete ones should be encoded by dummies.

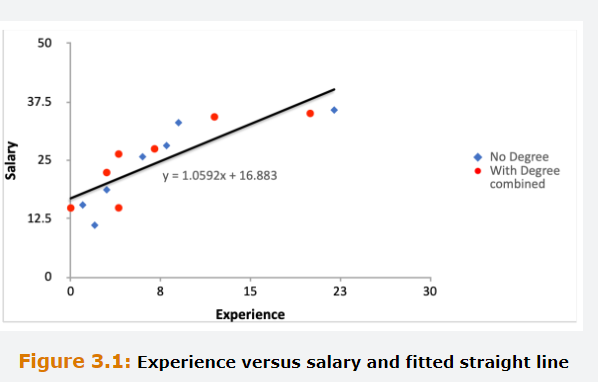


A screenshot of a table

Description automatically generated

In this example, we believe that salary should be driven by experience. So, to begin with, we run a simple linear regression of salary (y) on experience () , and we would obtain the following:

We use hat above the output variable to denote the fitted equation form the regression line. Here, the intercept estimate is 16.88, meaning that someone just joining the bank with no experience could expect to earn 16.88 per hour on average, and each additional year of experience would lead to an average salary increase of 1.06.

After a while.

The plot suggest that the line fits the data well, although it seems that additional years of experience lead wages to increase significantly when experience is low but the impact tails off after a while.

It appears from the diagram that each additional year of experience leads to a lower incremental increase in wages.

We cannot capture this with a linear model, because it embodies a fixed gradient of the fitted line and therefore a fixed relationship between x and y whatever the value of x.

In order to capture the potential non-linearity in the relationship between ages and experience, we can fit a quadratic model, where the squares of experience is included in the model as a second feature.

The fitted model is now:

A graph with red and blue dots

Description automatically generated

Note that when the square of experience is added, the remaining parameters estimation change compared to their valuer when only the level of experience was included on the model.

Know that we have two features relating to experience, determining the relationship between experience and salary is slightly trickier.

It is clear that when the square of experience is included, the relationship between salary and experience is no longer constant with respect to experience.

Setting the expression to zero and rearranging it to make experience the subject of the formula will show the value of the feature that maximizes expected salary, which is 29.6 years.

The regression model now includes two variables, experience and its squared value, but the degree feature can affect the salary as well. This is captured by a dummy variable taking the value 1 if the person has a degree and zero if they do not.

The 0.92 should be interpreted as an employee with a degree could expect to earn an additional 92 cents per hour on average compared with someone having identical experience but no degree.

If we had more than one dummy variable in the model to capture other qualitative information, we would interpret the associated parameters the same way.

## **3.1.3. Potential problems with Regressions**

In some cases, even if we can estimate values for the parameters, these ones might no longer be optimal or reliable.

**Problem 1** is the use of wrong features or wrong functional form.

On the previous example, we assumed that the most appropriate features to model variation in wage rates for bank employees were based on their experience and degree level qualifications.

Nevertheless, it might be that the model did not include the most relevant features, and broadly there are three ways that the model can be wrong.

**Firstly,** the model omits some relevant features, which could occur if the true relationship describing the output includes some extra features that the researcher has not included in the model, due to lack of data or unawareness of their relevance. This could be a serious misspecification that could lead the parameter estimates to be biased and not become more accurate as the sample size increases.

**Secondly**, the model includes some irrelevant features, this is less serious than the first misspecification, but can result in inefficiency where the parameters are not estimated precisely. The model will find out to be hard to generalize from the specific training sample to the test data.

**Thirdly**, the model includes the correct features, but they are incorporated in the wrong way. This is known as an incorrect functional form. It could occur, for instance, is the true relationship between the features and the output is non-linear but a linear regression model is used.

These three problems are hard to resolve in practice than they appear, because the researcher never knows the true relationship between features.

The remedy for this is Strong theoretical knowledge of the problem at hand and the wider context can be valuable in guiding the model development, rather than a purely data driven approach.

**Problem 2** is multicollinearity which occurs when the features are highly related to one another.

Perfect multicollinearity occurs when two or more of the features have an exactly linear relationship that holds for every data point. The solution is to remove one or more of these perfect correlated features from the model.

Near multicollinearity occurs when two or more features are almost perfectly correlated. A common consequence is that the parameter estimates become highly unstable, changing wildly when a feature is added or removed from the model. The remedies include removing one or more of these features or turning them into a ratio/difference rather than including them individually.

**Problem 3** are Outliers, which are anomalous data points that lies a long way from the others. These ones can even a high impact on the parameter’s estimations.

The least square technique used to estimate the parameters in a regression model takes the sum of square of distances from the points to the fitted line and the process of squaring these distances means that points that are a considerable distance from the others will exert a disproportionate effect on the estimates.

A graph of a function

Description automatically generated with medium confidence

Outliers can be detected by examining a plot of the residuals, the difference between the actual data points and the corresponding fitted values from the regression line, and noting any points that lie further from the line than others.

Note that if both the input and output values are further from the other datapoints but the point nonetheless lies near the regression line, this would not be classified as an outlier.

A more sophisticated method is to calculate the Cook’s distance, which measures the influence of each individual data point on the parameter estimates.

This is achieved by removing each data point separately from the regression and determining the difference in model fit for all the remaining data points. If a particular data point is not very influential for parameter estimation (hence is not an outlier), the model fit will not be changed by a lot and the Cook’s distance will be small.

**Problem 4** is heteroskedasticity, meaning that the variance is not constant, and occurs frequently in time series data.

Can lead to several issues with regression estimation, most notably that it becomes inefficient and that it is hard to accurately evaluate the empirical importance of each future for determining the output.

As for outlier detection, a residual plot can sometimes be useful in detecting heteroskedasticity, where we would be looking for whether the spread of the residuals around their mean, usually of zero, is constant or systematically changing.

There are also some statistical tests for Heteroskedasticity, such as Goldfeld-Quandt test, which splits the sample into two parts and statistically compares the residual variances between the two.

Alternatively, the White’s test involves obtaining the fitted values, , from a regression and conducting a second, auxiliary, regression of the squares of the fitted values on the squares of the features and interactions between them. If there is no heteroskedasticity, the parameter estimates from this auxiliary regression will not be statistically significant.

A graph with blue dots and red line

Description automatically generated

Several remedies for Heteroskedasticity are to weight the observations to account for the changing error variance using a technique known as weighted least squares (WLS) instead of OLS. Alternatively, making a logarithmic transformation of the variables and using them in place of the raw variables.

## **3.1.4. Stepwise Regression Procedures**

As discussed above, the presence of non-informative or redundant features in linear regression model can add uncertainty to the predictions and reduce the effectiveness of the model, especially in the presence of highly correlated features.

This introduces the need to remove non informative predictors.

Stepwise regression, like LASSO regularization technique is a method for feature selection. It belongs to the category of wrapper methods, which add or remove predictors to a regression with the aim of finding the combination that maximizes the model performance.

The inclusion or exclusion of features is based on criterion that measures the predictive accuracy of alternative set of predictors. A popular approach is to choose the model that minimizes the Akaike information criteria (AIC) which is a measure of prediction errors adjusted to account for the number of features in the model.

Unlike R squared, AIC penalizes large models and therefore it can either increase or decrease when an additional feature is added to the model.

There are various stepwise procedures, but he most straightforward are the unidirectional forward and backward stepwise selection methods.

The forward starts with a model with no features and includes additional features into the model one-at-the time starting from those that produce the largest drop in the AIC. The procedure stops when the addition of any new predictor fails to decrease the AIC.

The backwards begins with the full model and removes the predictors one-by-one starting with the least important, until any further elimination fails to decrease the AIC.

A screenshot of a screen

Description automatically generated

As it emerges from the above example, the two procedures will not necessarily select the same model.in fact, more generally, there is no guarantee that either of the two methods will select the optimal model, as only a subset of the models, in which m is the number of predictors, is considered.

Although in principle it is possible to estimate all the possible models and compare them, this is very impractical when the set of candidate predictors is large. Therefore, when the set of predictors is large, bot forward ad backwards stepwise regression offers a valid alternative.

Both procedures have their pros and cons and the choice between the two depends on the problem at hand. Backwards selection tends to be more computationally efficient when the set of candidate predictors is large. However, as the full model is the first to be estimated, the number of predictors is required to be strictly smaller than the number of observations. In contrast, forward selection can also be applied when the number of predictors is larger than the number of observations.

## **3.2. Classification Problems**

In finance, a lot of times, the model output, dependent variable, is categorical and where the output for each observation can only be one of a small number of categories.

In the Machine learning jargon, the problem of prediction a qualitative outcome is defined to be a classification problem and assigning an observation to one class rather than another is referred to as classifying the observation.

In binary situations, is interesting to model the probability of one of the outcomes occurring. One outcome, referred to as the positive outcome, is assigned a value of one, and the other, the negative one, a value of zero.

A Standard linear model would be inappropriate to apply in such cases because there would be nothing in the model’s design to ensure that the estimated probabilities lie between zero and one, and we could obtain nonsensical predictions.

## **3.2.1. Logistic Regression**

This specification uses a cumulative logistic function transformation resulting in the output being bounded between zero and one.

The logistic function has a sigmoid shape and is written by

When there are m features, the functional form is estimated as,

And the probability that is given by:

And the probability that is (1-)

**A graph of a function

Description automatically generated**

The parameter estimates from a logit model cannot be interpreted in the usual fashion due to the presence of the logistic transformation, which is nonlinear. Nevertheless, their signs and levels of statistical significance can still be examined.

From an example of credit lending, in which 0 is not to default and 1 is to purely default, the borrowers with longer loan terms and those paying higher interest rates have a significantly higher probability of default, whereas those with a mortgage have a significantly lower probability of default.

A white and orange text on a white background

Description automatically generated

## **3.2.2. Other Types of limited Dependent Variable Models**

Discrete choice models are based on models that try to predict a categorical, multiple choice, dependent variable.

The Classical example is where a commute chooses between different transport types and we are interested in modelling the probability that a particular individual will travel by each mode, which would be the output.

As per the Binary output It would not be appropriate to use linear regression model, we should use an extension of the logit regression describe on the previous chapter, known as **Multinominal Logit Models**.

We estimate models for all categories but one, which will serve as a baseline category. As per the above example, we could model the probability that a commuter will choose to travel by car, and the probability of traveling by bike. Then the probability of traveling by bus is simply one minus the sum of the two calculated probabilities.

**Ordinary variables models**, distinct from the above because here the categorical variables have implicating ordering, such as credit ratings, risk scales from 1-10 and so on.

For modelling ordinal variables where there are more than two outcomes, ordered logit models would be used. The estimation principles are the same as for the binary case, but the values of cutoff parameters between categories must also be estimated.

In some circumstances, the values of the output variable in a model that we observe are not sampled randomly from the population, resulting in a biased sample.

## **3.3 Linear Discriminant Analysis**

When there are multiple, well-separated classes, the estimates from a logistic regression turn out to be very unstable. In this case, an alternative to the logistic regression is offered by the LDA.

LDA assumes that the Joint distributions of features is multivariate normal with a common variance-covariance matrix, but with different mean vectors. Like logit, the idea is to assign each instance to the class with the highest conditional probability.

A discriminant function is calculated for each of the classes. It is the probability that a new data point belongs to that class. New data points are classified based on which class has the highest probability. It is possible to show that the discriminant function are linear function of x for each class j, for j=1,…,g.

Where x is the feature vector, is the vector containing the means of the predictors of each g classes, estimated using the training sample data, is the inverse of the data covariance matrix, is the prior probability of class j, and T denotes the transpose operator.

The prior probabilities could be assumed to be equal for all classes or estimated using the frequency of class j in the training data. A new data point is assigned to the class for which is the largest. LDA has proven to work greatly in practice, even when the assumptions are not met.

As an example, consider a sample of 10 borrowers to be classified as defaulted or not default. Because LDA, like PCA, requires data scaling, the data provided in the table have been already standardized.

A screenshot of a cell phone

Description automatically generated

A value of 1 means that it defaulted, 0 otherwise.

A fist stage is to calculate te class means for each predictor from the data by averaging the observations that belong to each class.

The covariance matrix can be estimated as,

Where the variance of both predictors is equal to 1 as the predictors have been standardize. The inverse of this matrix is,

Finally, we can set the prior probabilities to 0.7 and 0.3, based on the frequencies of each class in the training sample.

Suppose now that we have to classify an applicant with a standardize balance equaling to -1.42 and a standardize income equaling to -0.2. This means that we can write

For Default we would have,

And nondefault:

Because nondefault is higher the applicant is nondefault.

## **Appendix 3.A The Heckman 2 stage Procedure**

The values of the output variable in a model that we observe may not be sampled randomly from the population, which leads to a biased sample.

For instance, the willingness to respond to a survey is often correlated with the variables we are trying to measure. So if we asked bank costumers to complete the survey indicating their satisfaction with the level of service that they have received, those who are the least happy may be those most likely to complete the survey.

If we were then interested in modelling the factors that affect costumer satisfaction levels using these survey results, our parameter estimates would be biased.

Another situation where such an issue would occur is in the context of modelling the value of share repurchases. Most listed firms do not make share repurchases, and therefore the output variable would have some problematic characteristics: no observations would be negative, and the bulk of observations would have a value of 0, with the remainder having a distribution a long way from 0.

The Heckman approach deals with these situations by separating them into two stages.

Firstly, model the probability that a bank costumer will complete a survey or firm will choose to make share repurchases using a binary logit function.

Secondly, model the determinants of costumer satisfaction among those who have elected to complete the survey or model the size of the repurchase among firms that have chosen to make them.

## **Appendix 3.B Fisher Discriminant Analysis**

Is an alternative to the Linear discriminant analysis (LDA). The idea is to find a vector on which to project the data such that the maximum between group variance of the projection relative to withing-group variance is obtained.

In this respect, LDA can be related to dimensionality reduction techniques. We find a projection on the data on a lower dimensional space that is optimal for classification of the data. Once this vector have been found, new data to be classified are projected onto this vector and assigned to the class whose mean they are closer to.

In other words, we aim to find a way to separate the data into g distinct classes such that the distance between the means of different classes is maximized while the variation within each class is minimized. Technically, we find the vector b that maximizes the signal-to-noise ratio, which is given by the ratio of the between and within group variances.

Where B is the between Group covariance matrix and W is the Within group covariance matrix.

To make it concrete, consider a simple case where only two classes are available. In this case, solving the maximization problem gives:

Where , and are the covariance matrix, and the class means respectively. The discriminant vector is perpendicular to b.

Therefore, the discriminant score of a new data instance x is obtained by projecting x onto vector b:

We classify x as belonging to class 2 if , in which c is a threshold to be decided by the researcher. If we are ready to assume that the two classes display approximately the same distribution, then the optimal threshold is given by:

In this case, the Fisher formulation of the problem is equivalent to the probabilistic formulation that was presented earlier for LDA with equal prior probabilities.

## **Appendix 3.C Linear Discriminant Analysis Example**

Data

A table with numbers and words

Description automatically generated

Subscribed is the outcome of the campaign, Balance is the outstanding balance of the Client at the bank in US Dollars and Age is the age of the client in Years.

Perform the following Tasks:

A. Standardize the features.

B. Estimate the within-group means for the clients who subscribed (class 1) and those who did not subscribe (class 2).

C. Compute the data covariance matrix

D. How would you classify a client who is 51 years old and has an outstanding balance of 10,635 Dollars?

**A.**

To standardize the features, we first must compute their sample mean and standard deviation. The sample mean of the first feature is and its standard deviation is . The sample mean of the second feature is and its standard deviation is .

The Standardized features are obtained from subtracting the mean and divided by the standard deviation.

A white rectangular box with black text

Description automatically generated

**B.** The within groups mean are simply the means of the features for each class. For Class 1, the vector of the features means would be

Where the first is obtained by averaging the Class “yes” standardized values of Balance and the second the standardized values of age.

Same thing for second class to obtain

**C.** As the two features have been standardized, their variance are both equal to one. Therefore, we only need to compute the covariance between the two features, which is given by:

Hence, the covariance Matrix is:

**D.** The only further piece of information needed to determine is the probability of each class. The probability of yes is 0.6 and 0.4 is the probability of no. The inverse covariance matrix can be easily obtained by hand or using the function MINVERSE in Excel.

The standardize features of the new data point are:

Therefore we get:

For Class 2

Since the Class 1 is higher, we predict to be class one. In fact, from the large value of standardized balance, 13.24, it is quite clear that the new client should be a subscriber.

## **Questions and Answers Module 2 Chapter 3 from GARP**

***3.1   
A. What is an Outlier in the Context of regression?***

An outlier is a data point that demonstrably does not fit with the patter of the others so that in a regression context, the fitted and actual values would be a long way apart, leading to a residual of larger magnitude than the others.

***B. How Can outliers be detected?***

There are various methods available to detect outliers. A good first step is to examine the residuals from the purposed model to see whether any are significantly larger in absolute value than the others. More formally a measure known as Cook’s distance can be calculated for each point. This evaluates how much each parameter would change if a given data point were excluded from the sample. Large Values of Cook’s distance indicate a point that would be more likely considered an outlier.

***3.2 What is a stepwise regression and how does it work.***

Stepwise regression is a technique for feature selection. Beginning with a list of candidate features that could be included in the model, the analyst selects an approach: Either beginning with a model containing no features (forward selection) or containing all the features (backwards selection). With forward selection, the analyst adds the feature that would have the most additional explanatory power until a further addition does not decrease the AIC. With Backwards selection, all features are initially included in the model, and they remove one-by-one starting with the feature having the least explanatory power until removing a further variable fails to decrease AIC.

***3.3 Explain why linear Regression cannot be used when the dependent variable in a regression model can only take the values 0 or 1.***

Linear regression is used to fit models that use numerical data that is continuous. But here the target variable is a binary value. When linear regression is used, there is nothing in the estimation process that would ensure the fitted values from the regression model would lie between 0 and 1. Truncating the fitted values to 0 and 1 would be inadvisable as the result would be too many values at these extreme points.

***3.4 Explain the main assumptions underlying LDA.***

The Standard approach to Linear discriminant Analysis assumes that the data arise from g multivariate normal distributions with different mean vectors but common covariance Matrix.

***3.5 Exercise***

Suppose that we have the following results for a model estimated using ordinary least squares for the relationship between a firm’s return on Assets (ROA) and its SIZE.

Where ROA is measured in percent and SIZE is measured in $M.

***A . According to this model, is the relationship between SIZE and ROA Linear?***

No, because the equation includes a squared term in SIZE, the relationship between SIZE and ROA is non-linear, therefore the relationship between the two variables will depend on the value of SIZE.

***B . Comparing two firms with market caps of 100M and 110M, what would be the difference in ROA?***

Best way is just to replace SIZE with the values, calculated ROA and the Differences between both SIZE values. It would be 0.14% higher for the 110M one.

***C . What would be the optimal size of firm for an investor to choose, if they wanted a firm with maximal ROA, and what ROA would this generate.***

We need to set the derivative to zero, and rearrange for SIZE to give the value of SIZE that Maximizes ROA:

Therefore, a firm with market cap of 160M$ would have the highest possible value of ROA. To calculate the latter, we simply plug 1.6 into the original fitted equation given:

## **4.0 Supervised Learning part 2: Machine Learning Techniques.**

Learning Objectives:

Machine Learning techniques for classification and prediction problems, such as decision trees, k nearest neighbor and Support vector Machines. An overview of Neural networks.

Firstly, differentiate between the two types of decision tree and illustrate how each is constructed and interpreted.

Explain how pruning and ensemble techniques can be used to enhance the performance of decision trees.

Apply the K- nearest neighbor method for Classifications

Illustrate how support vector machines are used to clarify the data

Describe how neural networks are constructed and discuss associated challenges.

Discuss advanced neural networks structures.

Describe how autoencoders are used for dimensionality reduction and differentiate between autoencoders and PCA.

## **4.1 Decision Trees**

Are supervised machine learning techniques that examine input features sequentially. At each node is a question, which branches the observation into another node or a leaf (terminal node).

Although these ones are particularly popular for classification problems, they can also be employed to estimate the value of a continuous variable and so are sometimes known as classification and regression trees (CARTs).

CARTs are popular due to their interpretability, white box model, in contrast to other machine learning techniques such as neural networks, a black box model.

They tend to perform less well than other, black box, more sophisticated machine learning models.

To improve their performance, trees are often combined using ensemble techniques such as random forests, bagging and boosting.

A diagram of a credit score

Description automatically generated

## **4.1.1 Regression Trees**

The goal is to split the feature space into regions such that we minimize the residual sum of squares (RSS), given by:

Where is an observation in the training set, is the average outcome of the observations in region j, and J is the total number of regions.

Unfortunately, it is computationally infeasible to check all possible partitions of the feature space to find the one that minimizes RSS.

Therefore, we employ a top-down recursive binary splitting search. In this approach, we start with all the observations in one region and search for the split that produces the maximum reduction of the RSS. Then, for each of the two regions obtained in this way, we look for a further best split, and we proceed recursively until a given stopping criterion is reached.

For instance, suppose that we want to predict house price, based on age of the house and distance to the closest metro station.

A regression tree is a set of rules that tells us how we can optimally segment the training sample into regions of the feature space. If we estimate a tree for this dataset, we find that the first step is to split the sample between the houses that are less than 826.83 meters from the closest metro station.

Then we split this subsample between those that are smaller than 11.7 years, or greater than.

Region R1 from the below figure, contains all the houses that are below 11.7 years and closer than 826.83 m from the metro.

The prediction of the price per unit area for the houses in R1 is 52.25, which is the average price of houses belonging to that region of the feature space.

We continue to split the space into non overlapping regions until further splits fail to improve the prediction accuracy or some stopping criterion is reached.

A graph of datasets

Description automatically generated with medium confidence

A diagram of a tree

Description automatically generated

## **4.1.2 Classification Trees**

These ones follow a very similar logic to regression trees except that the outcome variable is categorical.

The objective is to split the data into groups that are as pure as possible, i.e., they contain the largest proportion of one class as possible). However, in classification problems, the RSS cannot be used as a criterion to determine the splits and we need to find a measure of purity. These ones are considered to be the Entropy and Gini Coefficient.

Entropy is a measure of disorder in a system.

Where J is the total number of possible outcomes, and is the probability of the outcome for j=1,…,J. Note that the formula includes the logarithm to base two rather than the LN.

The Gini coefficient is a measure of the impurity of a node and can be calculated as:

A small value of the Gini index indicates that a node mostly contains instances from the same class. Gini and entropy usually lead to very similar decision trees.

## **4.1.3 Classification Trees Example**

Suppose that a risk manager at an equity fund is concerned that firms held within portfolio will stop paying dividends next year and so whishes to build a model to predict whether a firm i will pay (=1) or will not pay a dividend (=0).

A model to classify this output is based on the variables from the table below. It’s a mixed sample of 20 non paying/paying dividend firms. For the outcome and binary values, the value 1 means a positive outcome.

The % of retail investors is a continuous variable that could take any real value from 0 to 100%. In a decision tree, we need to select a threshold value for each continuous variable that maximizes the information gain at a particular node.

Note that the optimal threshold will vary depending on the node at which splits occur.

Ideally, a particular question will provide a perfect split between categories, i.e. each terminal node will be a pure set. For instance, if it had been the case that no technology stocks paid a dividend, this would be highly beneficial information and the node containing technology stocks would be pure, only containing nonpaying dividend stocks.

On the other hand, the wors possible scenario would be exactly half of the tech stocks paid dividend, which would make this variable not as useful to be a discriminant factor.

A table of numbers and numbers

Description automatically generated

Looking at the output variable, 12 out of 20 firms in the sample paid a dividend.

Although is possible to construct the tree using entropy, we will use the Gini coefficient as the calculations are slightly simpler.

We measure the Gini coefficient before knowing anything about the features/independent variables.

This provides a base level with which we can compare the fall in the Gini Coefficient, which represents an information gain.

The first step is to select the feature that will go at the root node. This choice is made by selecting the one that would cause the Gini to drop the most, Large\_Cap.

First, examining the earning drops variable, among firms with an earnings drop =1, six paid dividends and four did not.

Similarly, among firms with no earnings drop, six paid dividend and four did not, having the same Gini Coefficient. We next calculate the average Gini coefficient for splitting according to this feature, which is calculated from weighting the coefficients for firms with earnings drop and for those without earnings drop according to the proportion of firms in each of those two categories.

We can calculate the information gained by:

This adds zero value, since the number of firms paying a dividend, both for firms that experienced an earnings drop and those that did not.

Repeat the process for the other 3 features.

Large\_Cap feature is 1

Large\_Cap feature is 0

Repeating the above for tech, information gained would be 0.003.

For the % of retail investors, because the variable is continuous, it is necessary to use an iterative procedure to determine the threshold that maximizes the information gain. It turns out that the information gained is less than 0.255 so Large\_Cap is the best choice for root node. Once this is done, the three branches out separately for the large\_cap firms (13 firms) and for those that are not (7 firms).

At subsequent nodes, features are chosen in a similar way to minimize Gini, and in this way to maximize the Information gained. Threes do not need to be simetrcial.

The three is completed when either a leaf is reached that is a pure set or all the features have already been used so that the data cannot be split further.

Creating a perfect classification tree is impossible in this example, so although some branches end up in a pure set others not.

A diagram of a company

Description automatically generated

## **4.1.4 Pruning**

Small trees offer several advantages over large trees:

Interpretability, fewer irrelevant features, and especially, avoidance of overfitting. As well as employing a separate testing sub sample, overfitting can be prevented by using stopping rules specified a priori, noted as pre or online pruning, pr by pruning the tree after is has been grown (post pruning).

A simple example of stopping rule is when a x number of breaches is reached, no further splitting is allowed. Another example is the termination of the splitting of a node if the number of observations under that node is smaller than a certain number.

Whereas Pre pruning prevents a tree from growing to much, post pruning consists of growing the tree fully and the identifying weak links’ ex post. In other words, it consists of replacing some subtrees with leaves whose label is the class of most of the instances that reach the subtree in the original classifier.

There are several pruning algorithms that can be distinguished between top-down and bottom-up approaches depending on whether they start at the leaves or at the root of the tree.

One of the simplest forms of pruning is reduced error pruning, which is a bottom-up approach. Starting ate the bottom, this algorithm replaces a node with its most popular class at any time that the resulting pruned tree does not perform worse than the original tree in the validation sample.

Another method in used to prune regression trees is cost complexity pruning. It consists of adding a penalty term to the RSS such that a trade off between accuracy over the training sample and the number of terminal nodes is established. The extend of the trade-off is determined by a tuning parameter alpha, which is chosen with cross validation.

## **4.1.5 Ensemble Techniques**

Trees tend to conduct worse performances than other qualifiers. A way to improve their performance is to ensembled a bunch of trees into a metamodel.

This has 2 objectives, being the first that through the “wisdom of crowds” and a result somewhat like the law of large numbers, model fit can be improved by making many predictions and averaging them. Second, the techniques can build in protection against overfitting.

*Bootstrap Aggregation*

Also know as bagging, involves bootstrapping from among the training sample to create multiple decision trees. The resulting predictions or classifications are aggregated to construct a new prediction or classification. A basic bagging Algorithm involves the following:

Firstly, sample a subset of the complete training set.

Secondly construct a decision tree on the usual fashion.

Thirdly, repeat steps 1 and 2 many times, sampling with replacement, so that an observation in one subsample can also be in another subsample.

Fourthly, if the problem is a regression, average across forecast of the several trees to obtain the final prediction. If the problem is a classification, record the class predicted by each of the trees and take a majority vote: the class predicted by most the threes is the overall prediction.

Because the data are sampled with replacement, some observations will not appear at all. The observations that were not selected ( called out of bag data) will not have been used for estimation in that replication can be used to evaluate model performance.

Pasting is an approach identical to bagging, except the sampling takes place without replacement, so that each data point is only used once.

*Random Forests*

Aggregating several forecasts work particularly well when the different leaners exhibit low correlation. Random forests provide an improvement over bagging by reducing the correlation across trees.

To achieve this, each time that a tree is split, only a random subset of all the features is considered. Although it may appear counterintuitive to purposefully exclude some features as each tree taken alone may result in a suboptimal result, it is a very strong predictor.

The logic is that, if, for instance, a feature is a very strong predictor whereas the rest only have modest predictive power, all the resulting trees will have this feature at the top and they are likely to yield very similar forecast. In contrast, by forcing some trees to deliberately ignore this strong predictor, the other features are given a chance, and the resulting forecasts are less correlated.

*Boosting*

Like Bagging, Boosting entail combining the forecasts from many decision trees. However, while in bagging each tree is grown independently from the others, in boosting each tree is grown while exploiting the information from the prediction errors of the previously grown trees.

The 2 main varieties of boosting are gradient boosting and adaptive boosting.

Gradient Boosting constructs a new model on the residuals of the previous one, which then become the target, i.e. the labels in the training set are replaced with the residuals from the previous iteration.

AdaBoost, involves training a model with equal weights on all observations and then, sequentially, increasing the weight or misclassified outputs to incentivize the classifier to focus more on those cases.

## **4.2 K Nearest Neighbors**

Is a simple, intuitive, supervised machine learning model that can be used for either Classification or prediction problems.

To predict the outcome or class, for an observation not in the training set, we search for the K observation in the training set that are closest to it using either Euclidean or Manhattan distance.

Our prediction is the mean of the nearest neighbor’s outcomes. If the problem is a classification one, the instance to be classified is assigned to the class which more neighbors belong to (majority voting).

KNN is sometimes termed as a Lazy learner because it does not learn the relationship in the dataset in the way that other approaches do. KNN does not build a model, instead, every time KNN encounters a new instance, it compares it to all the existing instances to make a prediction.

The steps to implement KNN are as follows:

Firstly, select a value of K and a distance Measure.

Secondly, among the points in the training sample, identify the K points in feature space that are closest to the point in feature space for which a prediction is to be made according to the chosen measure.

Thirdly, if it is a prediction problem, compute the mean of the outcomes for the K neighbors that have been identified, and if it is a classification problem, assign the instance to the class to which most of the nearest neighbors belong to.

On the example below, the Balance and Income were already standardized, Euclidean distance is the measure used and 1 means defaulted an 0 not defaulted.

A table with numbers and lines

Description automatically generated

Assume that we must classify a new instance, which is a borrower with a standardized balance of 0.9 and a standardized income of -0.61.

Suppose that we select K equal to four and we use the Euclidean distance to find the nearest neighbors.

We should compute the distance from the unclassified with all data points in the training data. For the first we would have:

Observations 4,8,9,10 are the closest ones. Therefore, the new instance would be considered as a Default one.

A crucial choice is the value of K. Small values of K tend to overfit the data whereas large values of K may underfit. A common choice is to set k approximately equal to the square root of N.

Another approach is to use cross-validation to tune K and choose the value that minimizes the error over the validation sample.

KNN is simple and yet tends to yield quite accurate forecasts.

However, its main disadvantage is that it is computationally intensive as the distance between one instance and all the others must be computed before KNN can identify the nearest neighbors.

Another drawback of this method is that it performs poorly when there are a few irrelevant or noisy features, as those can drive similar distances apart in feature space.

## **4.3 Support Vector Machines**

SVM are a class of supervised machine learning models that are particularly well suited to classification problems when there are a large number of features.

To understand how they work, we will start with a sample of linearly separatable data points that belong to one or two classes (labelled as -1 or +1). If there were only 2 features, this can be represented on the cartesian plane.

A diagram of a diagram

Description automatically generated

Our main goal is to identify the position of a line that would best separate the two groups, the classification boundary, enabling us to predict for an additional data point not in the sample whether the outcome should be -1 or +1. From the above picture, the blue and orange points can be perfectly separated either by using the dotted or solid line.

More generally, there is an infinite number of linear boundaries that perfectly classify the data.

Therefore, we need a metric that helps us to identify which of the boundaries is the most appropriate.

SVM uses a metric called margin. Broadly, the margin is the sum of the distances between the classification boundary and the closest instance in the training data for each of the two classes.

Given a classification boundary it is possible to construct two lines that are parallel to it and that touch the training data of opposite classes, and that have no points between them.

The training data points that are on those parallel lines are called the Support vectors, and the distance between the lines is the margin. The optimal classification boundary, a.k.a. the maximum margin classifier, is such that it is equidistant from each support vector and the margin is maximized.

A diagram of support vector

Description automatically generated

The maximum classifier creates a decision value D(x) that classifies a new instance such that if (Dx)>0 we would predict the instance to belong to class 1, otherwise , we would predict the instance to belong to class two.

## **4.3.1 Support Vector Machines Example**

A Bank to grant or not to grant loans.

A table of numbers and numbers

Description automatically generated with medium confidence

Loans granted are coded as +1 and loans not granted as -1.

The solution of the optimization problem leads to estimation of , and .

Therefore, the maximum margin classifier is :

And the margin constrains are:

If we had to classify a new borrower with Monthly income equal to 5.7 and total savings 3.5, we would obtain:

As this value is below 0, we would classify the loan as not granted, labeled -1.

A graph of a car loan

Description automatically generated with medium confidence

## **4.3.2 Support Vector Machines Extensions**

If the Model contemplated more than 2 features, instead of having a line in the center with the biggest margin, we would have a Hyperplane with several dimensions, one less than the number of features.

The two classes had a clear degree of separation on the example provided, but a more likely practical scenario would be that there is some overlap between the two.

In the cases that the data are not linearly separable, the approach described above could not be used and requires some modification.

A more flexible approach would be to use soft margins, which introduces a penalty term into the optimization for incorrect classifications.

## **4.4 Neural Networks**

ANN are a class of machine learning approaches loosely modeled on how the brain performs computation.

By far, the most common type of ANN is a feedforward network with backpropagation, a.k.a. multi-layer perceptron.

The basic unit of a multi-layer perceptron is the neuron, a unit that holds information. The neurons are arranged in layers and a multilayer perceptron consists of several layers of neurons.

A three-layer perceptron should include an input layer, a hidden layer and an output layer. Each input layer is connected to a hidden layer, and every connection have a weight attributed, , where the notation means that we are connecting neuron j in layer h with the neuron i in the next layer. Each of the inputs is multiplied by the weight associated with each link and passed to the hidden layer, activation function:

The term b is called bias and Is often added to the weighted sum of the features, being this one a constant, like the intercept on Linear regression.

A diagram of a network

Description automatically generated

The activation function introduces nonlinearity into the relationship between the inputs and outputs. Without it, the outputs from the model would merely be linear combinations of the hidden layer(s), which would, in turn, be linear combinations of the inputs. Such structure would be a linear regression, which is antagonist to the idea of neural networks, since the goal of the later is to discover complex nonlinear relationships.

The process of propagating the attributes from the input layer to the output is called feeding forward. A multi-layer perceptron can also contain more than one hidden layer.

Deep Learning refers to machine learning methods utilizing multiple neural network layers to extract the nonlinear relationships embedded in the data being modeled. The deep learning methods are widely used in natural language processing, generative artificial intelligence, image processing and so on.

## **4.4.1 The choice of Activation Function**

The logistic (sigmoid) function we encounter in connection with logistic regression.

This function outputs a value between 0 and 1. Other examples of activation functions are the softmax function, the rectified Linear Unit (ReLU), the leaky ReLU and the Hyperbolic tangent.

The softmax function is a more generalized version of the logistic activation function, that is used in multiclass classification problems.

For where K is the number of classes,

Softmax applies the standard exponential function to each element of the vector z of inputs and normalize the result by dividing by the sum of all exponentials. The sum of all f( is 1. This function exponentially magnifies the Importance of the largest member of the input value.

In contrast to sigmoid and Softmax, the ReLu activation function Is unconstrained from above. The ReLU activation function takes z as input and returns:

When the input is negative, it returns and zero and when is positive returns the value itself. The leaky ReLU activation function is a variation of the ReLU function which allows for small negative numbers:

Finally, the Hyperbolic activation function:

Has a similar shape to the logistic function but it produces values between -1 and +1 rather than zero and one.

A graph of a function

Description automatically generated

Notably, each layer can employ a different activation function, while all the neurons in the same layer apply the same activation function. However, it is common to use the same activation function for all the hidden layers. For the multiple layers perceptron and convolutional neural networks, a common choice is to use the ReLU function for hidden layers.

When a recurrent neural network is employed, popular choices of activation functions are the sigmoid and the hyperbolic. The activation function of the output layer tends to depend on the problem at hand. The sigmoid is a common choice for binary problems, as the softmax is typically employed in multiclass classification problems.

## **4.4.2 A numerical Example**

Example of a basic neural network. There are Three layers, the input the hidden and the output one.

The input one contains four neurons, as much as the number of features., in which

The output layer is a binary classification to determine whether each observation belongs to 0 or 1.

In the hidden layers there are only two neurons.

For now, assume the weights as follows:

The biases are and and the activation function for the hidden layer is ReLU.

The value of is obtained via multiplying the weights by the features and summing their values and the bias.

0.15\*0.6-0.4\*(-0.4)-0.3\*0.3+0.2\*(-0.2)+0.7= 0.82

Then, the chosen activation function is applied to obtain

Similarly, is calculated as:

-0.2\*0.6+0.3\*(-0.4)-0.6\*0.3-0.2\*(-0.2)+0.3= -0.08

Which is transformed to:

In the output layer there is one neuron, and the activation function is logistic, by assuming that is binary.

The weights are , the bias is

The value of the Output neuron, our prediction, is obtained by applying the logistic activation function to the weighted sum of

This can be interpreted as the probability of y is 1, meaning that since 0.65 is higher than 0.5, the observation is assigned to the 1 class.

A diagram of a network

Description automatically generated

## **4.4.3 Backpropagation**

The set of weights that link the neurons are parameters. These were given in the previous example, but they must be estimated. The strategy is to find such weights that some measure of classification or prediction error, a loss function, typically residual sum of squares or mean squared error, is minimized.

This procedure is recursive, at the beginning the weights are assigned random values and then a first training example is forward propagated to the networks output.

Then the weights are updated using the error between the calculated and actual values of the labels. After updating the weights, a new example is introduced, and the weights are updated again. When the last training example is introduced, one epoch, i.e. iteration is completed.

In general, the successful training of a network involves many epochs and therefore neural networks are usually regarded as a computationally intensive technique.

## **4.4.4 Architectural Issues**

Neural networks are an incredibly flexible tool. It was proved empirically that with the right number of hidden neuros, and under some assumptions, neural networks can approximate any function with arbitrary precision. This result is known as the universal approximation theorem.

However, in practice, neural networks are very prone to overfitting the training data and this problem is worse with large networks.

Usually, one starts with one or two hidden layers and add more layers after assessing the performance of the model. Adding more layers is useful for modeling more complex phenomenon. This leave us with the question on how to determine the current amount of hidden layers and the number of neurons in which layer.

One solution could be to experiment with different sizes of network, compute their accuracy over the test sample and pick the network structure that minimizes the error rates.

Consider a network with 100 features, one hidden layer with 100 neurons and an output layer with only one neuron. The estimated weights would be 100\*100+1=10,001. Besides, many epochs are generally needed to reach convergence, which makes this approach highly impractical.

An alternative is to find an appropriate size of network proceeds. We start with a small number of hidden layers. At the end of each epoch, the algorithm computes the value of the loss function over the training set. This value is likely to keep decreasing when the number of epochs increases, until a point is reached where the performance fails to improve. This can happen wither because the network lacks the necessary flexibility to make correct predictions, or because a local minimum has been found.

When this is observed, additional layers are added to the network and the training is resumed. If this allows for a reduction in the value of the loss function, the newly added neuros are retained. If not, the smaller model is preferred.

The number of neurons within each layer is dictated by the size of the features set and target. It was common practice to structure the network such that the number of neurons decreased from one layer to the next as the network approaches the output layer. This pyramid style structure has been largely superseded by a more uniform structure with an equal number of neurons in the hidden layers coupled with regularization techniques to ensure that the model is not overfitting.

## **4.4.5 Overfitting**

When the model is large and issufiicient training examples have been provided, neural networks tend to learn random artifacts of the training data. This implies that they will fail to generalize well to unseen test instances.

An extreme form of overfitting is memorization, which results in an almost perfect fit to the training data, and which is not uncommon with neural networks.

Overfitting signs are:

Firstly, the same model obtains very different predictions depending on the sample is it trained with.

Secondly, the gap between the prediction error over the training and the test sample is very large.

Apart from avoiding parameter proliferation, there are a few techniques that can be used to reduce overfitting.

Penalty based regularization involves imposing a penalty over the loss function, such as , where d is the number of neurons.

Dropout is an ensemble technique explicitly designed for neural networks. It consists of creating alternative networks by selectively dropping a few neurons each time. The forecast from these networks are then aggregated to create the final prediction.

Early stopping, in which the optimization is stopped before converging to the optimal solution on the training data. A portion of the data is held out and used to determine the optimal stopping point. The training is stopped when the error in the hold out samples starts to rise.

## **4.4.6 Advanced Neural Network Structures**

*Convolutional Neural Networks*

CNN, are a specialized form of Neural network where the neurons in one layer are only connected to a subset of neurons in the next layer. They are designed to work with inputs that have a grid structure and where adjacent points in the grid exhibit dependencies.

The most obvious application of CNNs is with 2 dimensional images, but they can also be employed for textual, voice or time series data.

CNNs are ideal in such cases because the number of network weights to be trained is drastically reduced, which results In faster training of the model. The most common type of convolutional layer is the 2D or planar convolutional layer.

This one applies an n\*n kernel matrix W over a m\*m input grid to obtain a new filtered image that has a smaller sizer than the original image. This new one is called feature map.

The kernel matrix contains weights that should be learned during the training process, but in the following example, these ones are given for simplicity.

Input is a 4\*4 image, the matrix X:

This can be filtered using a 3\*3 Kernel:

The feature map is obtained by sliding the kernel over the image starting from the top left corner to move the kernel through all the positions where it fits entirely within the boundaries of the image.

Each position corresponds to a single cell in the feature map, the value of which is calculated by multiplying together the kernel value and the underlying image for each of the cells in the kernel, and then adding all these numbers together.

In practice , we will start by replacing the are on the top left corner.

A diagram of a mathematical equation

Description automatically generated

The feature map should be:

Then, we slide to the right:

A blue square with numbers and a black circle

Description automatically generated with medium confidence

The down:

A math equation with numbers and symbols

Description automatically generated

It is now intuitive to assume the final position as

The area in red, blue and green are termed a receptive field. It is the region in the input space that influences a cell in the feature map.

A non-linear layer, where a nonlinear activation function is applied to the feature map, can also be added after the convolutional layer.

It is also common to have a pooling layer. The pooling layer replaces the output of the previous layer at certain locations with summary statistics. For instance, it would be possible to summarize F by taking the average or the maximum of the values in the cells.

CNNs are parsimonious in terms of parameters as the same weights are applied to all receptive fields. Therefore, CNNs are useful to process images, which typically involve millions of pixels.

*Recurrent Neural Networks*

Differ from a Standard multilayer perceptron as the former models employ a temporal sequence to preserve the order in which the observations occur. In other words, RNN is designed to have some memory. RNNs are often employed in time series applications, and they are at the heart of large language models.

## **4.5 Autoencoders**

Are a class of Artificial Neural Network Models (ANNs) used for unsupervised learning. They are feedforward specifications, but the outputs are the same features as the inputs, and hence there are no labels.

Unlike K-means clustering, autoencoders are primarily used for dimensionality reduction and so are best thought of as non-linear extensions of PCA.

Autoencoders can provide a compact representation of the feature data and are particularly useful for high dimensional systems.

It should be noted that although PCA is commonly discussed in machine learning Contexts, in fact there is no learning involved, because it is merely a decomposition with a unique mathematical solution.

Autoencoders, on the other hand, are trained to learn the relationships present in the Data through model estimation.

The advantages of auto encoders over PCA is the use of non-linear activation functions which provide the universal approximation property in high dimensional space.

The features are put through the encoder, which is a function, to arrive at the values on the hidden layer. Then, the values in the hidden layer are converted back to the feature values through the Decoder.

The optimization objective is to reconstruct the original features as accurately as possible. The weights between the input layer and the hidden layer encode the information from the features, and the weights between the hidden layer and the output layer decode the information.

When the number of neurons on the Hidden Layer is smaller than the number of features, this is called a constricted or bottleneck hidden layer and leads to dimensionality reduction.

A diagram of a network

Description automatically generated

The hidden layers are simply calculated as a weighted sum of the inputs, and the outputs (reconstructed features) are a weighted sum of the value on the hidden layers.

The values on the hidden layer can be calculated as:

Where is the bias term.

The values at the hidden layer are the reduced dimension representation of the data, also known as the code.

If we let denote the fitted output, i.e. the reconstructed values from the model this would be calculated as:

The weights are chosen by minimizing a loss function, L, akin to the Residual Sum of Squares (RSS) in a linear regression.

An alternative would be mean Squared error (MSE)

L will be a positive number to reflect that the feature inputs will no be reconstructed precisely after the encoding and decoding processes, but this is the price paid in order to obtain a more parsimonious representation.

To this point, the autoencoder is linear in the weights, and thus it will perform a function comparable to PCA.

More specifically, in such a linear model, if there is only one hidden layer with K nodes, if both the encoder and decoder are linear, and if the inputs are suitably normalized, then the encoder hidden nodes will be the first K principal Components.

However it is more common to use a non-linear autoencoder by applying an activation function to the weighted sums in the hidden layers. The activation function, as previously discussed, introduces nonlinearity into the relationship between the inputs and outputs.

Without them, the outputs from the model would merely be linear combinations of the hidden layer(s), which would, in turn, be linear combinations of the inputs.

The new value of the hidden layers would be :

Where phi is the activation function.

By capturing these nonlinearities in the data, activation functions can also allow the number of nodes in the hidden layers to be further reduced so that the representation is even more compact than if a purely linear specification was used.

The number of hidden neurons is usually lower than the number of features to allow dimensionality reduction. If the numbers were the same, it would be possible to trivially reconstruct the exact features, but this would be pointless since there will not be any dimensionality reduction.

If the value of hidden Layers is higher than the number of features, we would have a sparse autoencoder. Hence, eventhough the number of hidden units is large, many of the weights, are set to zero, so that the effective number of weights is much lower and commensurate with a smaller number of hidden units/neurons.

By adding additional hidden layers, we form a deep autoencoder, which has more scope to capture more sophisticated nonlinear patterns between the features. These ones tend to be symmetrical to the center hidden Layer.

We evaluate autoencoders by calculating the Loss of the final fitted model, which is called the reconstruction error.

A table with numbers and symbols

Description automatically generated

## **Appendix 4.A Technical details of how SVM are Determined**

Considering a two-dimensional case with features x1 and x2. Upper margins constrain is defined as:

With the lower boundary:

Where w are the weights, l- and u+ superscripts denote the data points corresponding to the lower and upper margin respectively, and x1 and x2 denote features 1 and 2.

Subtracting the second of these two equations rom the first provides us with an equation for the margin width to be maximized:

Note that and are two specific instances of features.

We also need to apply two constrains to ensure that all the points lie on or outside of the estimated margins:

And:

The -,+ superscripts denote the data points corresponding to class 1 and 2, respectively, and N1 and N2 the number of observations in each class.

Using the index j to now denote all the observations regardless the class they belong to, we could combine the constrains as:

Where x1j and x2j combine the positive and negative outcomes, yi is +1 if the observation belongs to class 1 and -1 if the observation belongs to class 2, and N=N1+N2

By considering the relationship of and , through w1 and w2 one can show that the margin width is given by , where :

This is the Euclidean norm of weights. Maximizing the margin width is equivalent to minimizing . However, rather than working with , it is often easier to minimize because its derivative is just w. Therefore, we can write the optimization problem as:

To allow for Overlapping classes that are not linearly separable, we employ a hinge (max) function, which sets the penalty to zero for correct classifications and to distance between the point and the decision boundary for incorrect classifications. We can write the optimization problem as minimizing the function:

The function max() os zero for all correct classifications, however far the point is from the margin, but will be equal to the distance between the boundary and the point for incorrect classifications.

The regularization here is somewhat like a ridge regression, with the hyperparameter controlling the relative weight on the margin width versus incorrect classifications.

## **Questions and Answers Module 2 Chapter 4 from GARP – Machine Learning Techniques**

***4.1 What are the main differences between regression and classification decision trees?***

The main difference between classification and regression trees it the type of outcome variable. When the outcome variable is continuous, we refer to regression tress and when is categorical we refer to classification trees.

In both cases, we use a top-down recursive binary split to grow the three.

However, for regression trees the splits are decided to minimize the RSS, conversely, the splits in classification trees are decided to produce the largest drop either in entropy or the Gini coefficient.

***4.2 What are the main differences between bagging, boosting and random forest?***

All are ensemble techniques that are based on obtaining a final prediction as the average between the forecast of many trees.

When the problem is classification, a majority vote is generally used to determine the class to which an unsees instance is classified.

Bagging relies on the construction of many sub samples by randomly extracting observations from the training sample, with replacement. Tees are fitted on each of the sub samples and the forecast is obtained on the average (or the most popular class) across the predictions of those trees.

Random forests are like bagging, but the correlation among trees is reduced by using a random subset of p<m features at each split of the tree.

Finally, boosting is a sequential procedure where each new tree that is grown uses the information from the residuals of the previously grown trees. Gradient boosting and adaptive boosting.

***4.3 In the context of decision trees, what is pruning?***

Is a technique to reduce the size of the Tree to avoid overfitting and enhance interpretability. Pre or online pruning are conducted while the tree is growing by imposing a stopping criteria. Post pruning is conducted after the tree has been grown by replacing subtrees with leaves when the substitution does not decrease the predictive accuracy of the learner.

***4.4 What are the main advantages and disadvantages of decision trees vs other supervised machine learning techniques?***

The main advantage of Decision trees are their interpretability and the fact that they resemble the human decision-making process. They are called white box models.

Their Main disadvantage is that they are often less accurate than black box models such as neural networks. To enhance their performance, ensemble techniques are often uses, however this costs interpretability, there is a tradeoff between accuracy and interpretability.

***4.5 How would K nearest neighbors proceed to classify a new instance given a training sample of 10,000 observations?***

The first step involves choosing a measure distance, such as Euclidean or Manhattan and K, the number of nearest neighbors to be considered. For instance, we could sent K = which would be 100. Then, the distance between the instance to be classified and each of the instances in the training sample are computed and the K nearest neighbors are identified. A majority vote is used to assign the unclassified instance to the class of most of the neighbors.

***4.6 In the context of SVM, what is the maximum margin classifier***

The maximum margin classifier is the optimal decision boundary. It is the line (hyperplane) that is equidistant from the margin that constrains and maximizes the margin.

***4.7 In the Context of artificial neural networks, what is an activation function?***

An activation function for the neuron, popular choices are the sigmoid, the softmax, the ReLu and the Hyperbolic function.

***4.8 Describe some potential ways to address overfitting in a neural network.***

Overfitting is a common problem with neural networks, as their great flexibility is also their primary disadvantage. In addition to avoiding parameter proliferation, ways to address overfitting include penalty-based regularization, dropout and early stopping.

The fist approach involves imposing a penalty on the loss function. The second relies on the generation of alternatives networks by selectively dropping a few neurons.

The final forecast is obtained by aggregating the prediction from the various networks.

Finally, early stopping entails stopping the training before the convergence is achieved over the training sample. The stopping point is decided by looking at the error rate over a hold out sample.

***4.9 What is the primary purpose of autoencoders***

Dimensionality reduction, representing the most important characteristics of a dataset using a smaller number of transformed features.

***4.10***

***A. How do encoders achieve dimensionality reduction?***

By constructing a network with fewer nodes in the hidden layer that in the input and output layers.

***B. How do autoencoders differ from PCA in the way that they reduce the dimensionality of a dataset?***

PCA reduces the dimensionality of a dataset by constructing a set of orthogonal (linearly independent) components that are linear combinations of the orgitinal features. Although there will be m principal components if there are m features, only the first K of those, which are ordered to explain most of the variation within the features are retained.

Autoencoders use a neural network specification, usually with a non-linear activation function, which will identify a non-linear combination of the original features that is able to closely approximate them.

***4.11***

A diagram of a flowchart

Description automatically generated

***A.Describe the Structure of the Neural network***

Multi-layer perceptron with 3 layers, an input, a hidden and an output layer. The input layer contains three neurons and the hidden layer two neurons.

***B. Using the ReLU function at the hidden layer and the logistic function at the output layer, make a prediction for the default (=1) of a credit holder who is 35, has an income of 100.000 and an outstanding amount of 20.000. Biases are equal to zero.***

We shall first compute the value assigned to each the two neurons on the hidden layer.

Then applying the Sigmoid to the weighted sum of the inputs of the hidden layers

Because the predicted probability is high the risk that credit card holder would default is quite high.

***C. Why is the Sigmoid adequate for the output layer?***

Because is a classification problem. Values between 0 and 1, being 1 default and 0 not to default, and the value we obtain is the probability of such event to occur.

## **5. Semi supervised Learning**

Learning Objectives:

Explain how semi supervised learning differs from unsupervised and supervised learning.

Discuss the assumptions required for effective semi supervised learning.

Compare and contrast self-training and co training methos of semi supervised learning.

## **5.1 Introduction to Semi Supervised Learning**

When we have unlabeled data there are two possible approaches.

**Firstly**, either to remove the labels and treat the whole dataset as if is unlabeled, using appropriate techniques for unsupervised learning.

**Secondly**, drop the observations that are not labeled, and the use supervised learning on the remainder.

Neither of these options would be ideally because they both imply throwing away information. A third option could be to try to identify labels for the unlabeled data, but the labels could either be impossible to obtain or very costly to collect.

For instance, imagine that a bank asks its customers to write brief comments on the service they received. The bank would like to classify either as happy or unhappy and then correlate these classes with costumer specific info. This body of data would be unlabeled , because the comments would not have been classified yet, and creating labels would require a human to read through them individually and make a classification.

Doing this process could be infeasible if the data set is large and in other scenarios might require subject matter expertise. But in either case, should be possible to determine appropriate labels for a small subset of costumer, 30 for example.

This would create an hybrid data set with some labeled and some unlabeled observations. We could still employ one of the two approaches described above, but there is a third category of structure to consider, semi supervised learning also known as Weak supervision.

## **5.2.1 Semi Supervised Learning Assumptions**

Invariably, weak supervision is used in the context of classification rather than prediction scenarios. The technique makes use of parallels between classification and clustering and for it to work well several assumptions about the nature of the data need to hold.

Firstly, the **Clustering assumption**, i.e. the unlabeled data fall naturally into separable clusters (locally dense regions in feature space).

Secondly, the **Smoothness assumption,** or continuity assumption is assuming there is a smooth and continuous boundary separating the classes that can be used for deciding the classes of unlabeled instances.

Thirdly, the **Manifold Assumption**, i.e. the observed data point in the High-dimensional feature space are often concentrated along lower dimensional substructures that are topological manifolds. A topological manifold is a topological space that locally resembles the Euclidean Space . A way to understand the manifold assumption is to think about a sphere, 3D object, where all datapoints are concentrated on the surface (a two-dimensional object). The surface of a sphere I a two-dimensional manifold embedded in a three-dimensional space. The manifold assumption states that the input space is composed of many manifolds on which all the datapoints lie, and all the datapoints in the same manifold belong to the same class.

These assumptions imply that the clusters in the unlabeled data, map naturally onto the classifications on the labeled data.

For instance, a bank wants to develop a default model and had a dataset with two subjects, one with labels, i.e., if defaulted or not and a second subset without any lables where there is only info about the mortgage, but the bank does not know whether those customers defaulted or not.

The bank might build a classification model on the labeled part of the dataset and a clustering model for the unlabeled part of the dataset.

Weak supervision would work best if the clusters that formed in the unlabeled data naturally captured the same characteristics as the classification of the labeled data. For instance, two clusters, one with low income, high borrowing low collateral and the other the other way around.

If the different clusters and classifications separate the features in very different ways, the additional benefit from employing the unlabeled data to bolster the labeled data is much diminished.

This link between classification and clustering provides a foundation for how semi supervised learning works, specifically the assumption is that if a set of instances are clustered closely together, they would likely share the same label if they were labeled. On the other hand, points to far apart in feature space are less similar and therefore less likely to share a label.

## **5.2.2 Semi Supervised Learning Techniques**

There are two main techniques.

**Transductive** methods, which do not aim to build a generalize model and are therefore sometimes considered to arise from a “close world view”. In this case, because there is no model, the objective is solely to identify labels for the unlabeled data already observed. All instances need to be specified at the time of conducting the analysis, and no new instances can be incorporated into the study and classified at a later stage, so there is no separate test data.

One tranductive technique is label propagation, which is a graphical technique that assigns labels to unlabeled data based on how close they are to labeled data points using a metric such as the Euclidean Distance.

**Inductive** methods, on the other hand, involve building a model that links the features to the labels, and that can be applied to other instances. Common inductive methos include self-training and Co-training.

## **5.2.3 Self-Training**

The most popular due to its intuitiveness and simplicity.

It is sometimes referred to as a heuristic technique, because it employs unlabeled data from a supervised perspective, using models and methos for the latter, rather than using both labeled and unlabeled data together in learning. Self-Training is included in the Wrapper family.

**Firstly**, generate a classification model using any preferred technique (KNN, Logistic, etc) applied to the labeled part of the data.

**Secondly**, apply the model generated in the first step to all the unlabeled data and generate predicted labels for each instance in the unlabeled part of the data.

**Thirdly,** select the single instance for which the model’s predicted label has the highest probability of being correct based on the probabilities output from logistic regression, neural networks…

**Fourthly,** apply the predicted label to the instance selected at stage 3 and shift that datapoint from the unlabeled to the labeled portion of the data set.

**Fifthly,** return to stage 1 with the labeled set having now been enlarged by one observation and the unlabeled set reduced by one.

**Sixthly,** repeat stages 1 to 5 until all unlabeled data points have been labeled, the stop and that would be the final classification model.

These labels assigned are a.k.a **pseudo-label.**

The disadvantages of these are a couple.

Firstly, it is very **computationally intensive** because the model is retained as many times as there are instance in the unlabeled data. If computational resources are constrained, this problem can be mitigated by selecting the best predicted k observations at stage 3 and shifting all k observations, along with their predicted labels, in stage 4. For instance, if K=10, this will reduce by tenfold the number of rounds of training required.

Secondly, retraining the model after the addition of each individual datapoint can result in severe overfitting. Overfitting can be guarded against by a process known as co-training.

A diagram of a diagram

Description automatically generated

A self-training example, in which we have 4 unlabeled data points, where the borrowers default status is unknown. In this example we employ logistic regression for classifying the unlabeled borrowers into two classes, default or no default.

A table of numbers with orange text

Description automatically generated

Step 1 we fit the logistic model using only the labeled data points, i.e. 1 to 10.

Step 2, utilizing this trained model, we predict outcomes, the probability of default, and the probability of non-default for the unlabeled data.

Based on the results, the 13th borrower has the clearest split between the two categories, i.e. the biggest difference in the predicted probabilities.

A table with numbers and a number of objects

Description automatically generated with medium confidence

Step 3, we include the 13th observation in the labeled sample and label it as a default. Subsequently, we repeat the process using the updated data to fit the logistic regression model and predict the remaining unlabeled data.

This time the 11th borrower has the biggest difference in the predicted probabilities. Hence we include this observation into the labeled sample and label it as non-default.

We run the process again until obtain all labels.

## **5.2.4 Co-Training**

**Can be** applied when we have two different views of an example. Can utilize both views to build two classifiers that teach each other on unlabeled data.

Let us divide the feature ste x into two disjoin substest XA and XB representing two different views of the dataset. Co-training assumes that either XA or XB are individually sufficient to learning if we have enough labeled data and thus classifiers can be built for each of them.

**Firstly,** split feature set x into disjoints subsets XA and XB corresponding to two different views, A and B, both for the labeled and unlabeled data.

**Secondly**, generate classification models (model A and model B) for the two feature sets of the labeled data.

**Thirdly,** apply the models generated in step 2 to the two unlabeled subsets of data and generate predicted labels for each instance in the unlabeled subsets.

**Fourthly**, select the predicted observation from the unlabeled subset for each model with the highest probability score.

**Fifthly,** assign the predicted labels to the instances selected on point 4 and shift those from the unlabeled to the labeled sets. The key difference with from self-training is that the data points move to the labeled dataset of the other feature set. So the best predicted instance from unlabeled subset A moves to the labeled subset B and vice-versa.

**Sixthly**, return to stage 2 with the labeled sets having now been enlarged by one observation each, and the unlabeled sets reduce by one each.

**Seventhly,** repeat step 2 and 6 until all unlabeled data have been labeled, then stop and those would be the final classification models, one for each of the two disjoint sets.

**Eighthly,** estimate a single supervised model that reunites the two subsets A and B now that all instances have been labeled.

Because the Co-trainining use different subsets of features to build two different models to augment the training set, It reduces the risk of overfitting. Co-training is a.k.a disagreement-based method, because it exploits differences in the predictions based on the two subsets of features to improve the training classifications of both as they lean from one other.

A diagram of a diagram

Description automatically generated

Same example as in Self-training.

Initially, we separate the dataset into labelled and unlabeled samples. Subsequently we split the features into two distinct groups, A and B.

A table of numbers and symbols

Description automatically generated

Two Sigmoid modes A and B are trained independently using labeled data, focusing on each future group separately.

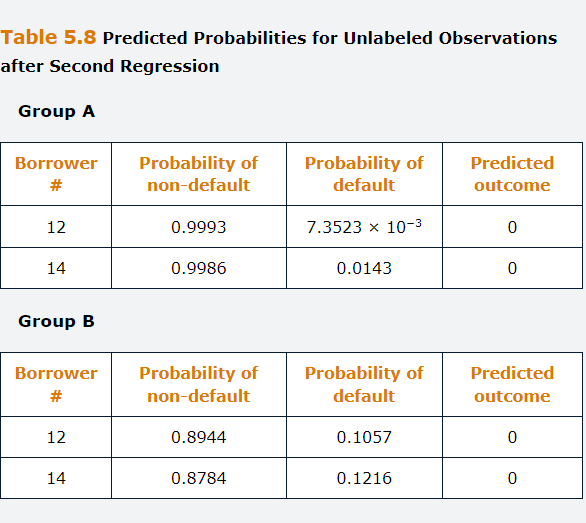
These models are then employed to predict the labels of the unlabeled observations.

A screenshot of a computer screen

Description automatically generated

Based on the prediction from the Sigmoids, the 13th borrower has the biggest difference in the probabilities in Model A, where as the 11th borrower has the biggest difference in the probabilities of model B. Consequently, the 13th observation is moved to group B and labeled has having defaulted, where as the 11th observation is move to group A with no default label.

Utilizing the updated labeled data, we repeat the process of fitting the sigmoid models and predicting the remaining unlabeled observations.



Since borrower 12th has the greatest difference in both models, it will be moved to both A and B group. Then we calculate the 14th.

## **5.2.5 Unsupervised Pre processing**

Involves working with the unlabeled data portion first before dealing with the labeled data in the subsequent stage. Within this family of approaches, there are at least three possibilities.

**Feature extraction**, which inlvoves employing techniques such as Principal Components analysis or autoencoders to reduce dimensionality of the unlabeled data and to represent it more efficiently.

**Cluster then label**, which is the combined unlabeled plus labeled datasets are subjected to clustering algorithms such as K-Means, and then resulting clusters are used to train a classifier model. If most of the labeled instance with a given label appear in the same cluster, then that label is assigned to all the unlabeled data points in the same cluster. This is another example of pseudo labeling.

**Pre training,** here the unlabeled data points are formed into clusters that are useful to develop preliminary decision boundaries prior to applying supervised learning.

## **Appendix 5.A Semi Supervised Learning Assumptions**

Its success relies on the clustering assumption, the smoothness or continuity assumption ad manifold assumption.

The figure below provides a pictorial representation of the first two assumptions. In panel (a) on the left the data belong to two classes, but fail to form clustes, and thus the clustering assumption is violated. The data points are dispersed in the feature space and do not form dense regions that can be separated.

In panel (b) on the right, the data form clusters, but these are partly overlapping and cannot be separated using a line, and the smoothness assumption fails.

**A screenshot of a computer

Description automatically generated**

The figure below shows the difference between manifold and non-manifold. The cubic box can be unfolded on a 2-dimensional space, and therefore each of its faces is a two-dimensional manifold. However, we will not be able to unfold the two cubes linked only by one hedge represented in panel b on the right in the same way as we did previously, meaning b is a non.-manifold structure.

Notably if each face of the cube in panel (a) contained datapoints that belong to a single class, the manifold assumption would be satisfied. However, if one face contained datapoints belonging to different classes, the assumption would be violated as the datapoints in manifold would not belong to the same class.

**A screenshot of a diagram

Description automatically generated**

## **Questions and Answers Module 2 Chapter 4 5 from GARP – Semi Supervised Learning**

***5.1***

***A- When should we consider using semi supervised learning?***

Is most valuable when the dataset contains both labeled and unlabeled instances, but where the labeled portion is relatively small and where it is infeasibly or excessively costly to manually add labels to the currently unlabeled instances.

***B- How does self-training differs from Co-training in semi supervised learning?***

Both are methods for applying pseudo labeled to the unlabeled part of the data in semi supervised learning. With self-training once the labeled data have been classified using a machine learning technique, the unlabeled data are assigned labels one at the time and added to the labeled portion of the data.

Co training adopts similar principals, but it involves splitting the features of the labeled and unlabeled data into two separate sub-datasets. Then, two separate classification models are built, and the most confident outcomes generated for each group of features become labeled instances for the other group of features.

## **6. Reinforcement Learning**

Is a machine learning technique that applies a trial-and-error feedback loop to train models to optimize actions that maximize a defined long-term or cumulative reward.

The output from reinforcement learning applications is a recommended action based on defined parameters rather than a prediction, classification or cluster produced in unsupervised or supervised learning.

Chapter learning objectives are as follows.

Explain the key principals and frameworks behind reinforcement learning

Compare and contrast exploration, exploitation and e-greedy strategies.

Describe reinforcement learning in the context of the Multi Armed Bandit (MAB) problem.

Explain Markov decision processes.

Differiantitate between the Monte Carlo and temporal difference methods.

Describe how neural networks can be used in reinforcement learning.

## **6.1 The Principles of Reinforcement Learning**

Is concerned with developing a policy for a series of decisions to maximize a long-term reward. In reinforcement learning, the learner is presented with feedback on the quality of the reward in a process analogous to trial and error. The technique is advantageous when decisions need to be made repeatedly so that the algorithm can learn based on the rewards or sanctions received in previous rounds.

Unlike both unsupervised and supervised learning, the output form reinforcement learning applications is a recommended action given the circumstances, rather than a prediction, classification or cluster.

A typical example of a situation where reinforcement learning can be applied would be a video game in which players can hone their strategies based on whether they won or lost, and by how much, in prior turns.

A further frequently employed game is that of teaching a dog to do tricks.

These ones are great examples because each one of the players/dogs behave differently, and therefore a fixed set of universal instructions covering all steps in the process cannot be developed.

For these instances is easy to define success but hard to specify a priori what is the appropriate action in every situation.

This technique has had very successful applications, in Chess and GO for instance. The algorithm learns by playing against itself many times and using a systematic trial and error approach.

More recent applications are on controlling movements in robots, self-driving cars, traffic light control, and inventory management.

There are many potential uses of reinforcement leaning in finance, including for technical trading, determining how to split a large volume of trades to sell quickly while minimizing the adverse price effect, and determining how much of a position to hedge using derivatives.

A disadvantage is that they tend to require larger amounts of training data than other machine learning approaches. The improvement function rises exponentially, starting poorly and then improving to the point of maximum optimization.

## **6.2. The Multi-Arm Bandit Problem**

This problem involves a gambler, agent, who can choose to play one pf several different slot machines. The gambler believes that the machines have different probabilities of winning, but is unsure which machine is more generous than the others.

The gambler objective is to maximize the total payout from a fixed number of rounds.

In each trial, the gambler picks one machine and the outcome, reward, is either that they win, or they lose. Therefore, each machine has its own probability distribution of rewards, and each round has only one step, i.e. one action, one state and one reward.

There is no other gambler/agent involved, i.e. it’s a single agent framework.

The gambler action in the current round does not affect the states in the subsequent rounds. Therefore, the current action only affects the current reward, not feature rewards.

## **6.2.1. Terminology in MABs**

Such models are determined in terms of states, actions and rewards. The states define the environment, i.e., the slot machine in each round, an action is the decision taken, decision on which slot machine to play, and rewards are the goal of the problem, payout from the chosen slot machine.

The aim is to choose the decision that maximizes the value of total subsequent rewards that are earned, possibly applying a discount rate to the rewards.

**Agent** is the person or algorithm making the decision. Usually there is a single agent, although in some models it is possible to have more than one, in which case the agents could be working together or in competition.

**Actions, A** are the possible choices that an agent can select from at each timestep. In MAB, the agent is free to choose which slot machine to play.

**State, S** are the circumstances or a description of the environment, in which the decision is being made at each time step. Because it is assumed that our action do not change the slot machines in any way and we are always playing on those machines, there is only one single state for our slots machines, and it does not change.

**Reward, R,** is the feedback that the agent receives based on its previous action. This could be either positive or negative, reward or sanction.

**Expected future rewards, G,** are the expected value of future rewards. The objective is to maximize this one. In our MBA case, the objective is to maximize the payout we get in the future, whenever we choose a slot machine.

**Policy,**  is the plan of action that the agent takes based on observing the current state. The policy maps the states to actions that will maximize the reward. Because there is only one single state in MAB problems, we only need to consider the policy in this state.

**Value function, V,** measures how good a state is. This is a.k.a, the state-value function. It relates the expected reward to a given state. It measures how good a state is. Because there is only one state in MABs, it is not relevant in this case, but it can be very useful for other problems with many states to guide the agent on policy improvement.

**Action-value Function, Q,** measures how good an action is, given a certain state, like our slot machines. This relates the expected reward to the actions and the state. It measures how good an action is, given a certain state. In the MBA context, it measures how rewarding it is if we choose a certain slot to play. This is very important and useful for us to compare the different actions and use them to optimize our policy.

The capital letters S and A are used to denote the set of states and the set of actions in general, whereas their lowercase counterparts denote specific states and actions. Time subscripts are generally suppressed unless they are specifically required for clarity, such as when describing the transition from the state in one period to the next.

**A diagram of a diagram

Description automatically generated**

## **6.2.2. Strategy in MAB**

A very intuitive way of playing is to always choose the best actions identified so far, which is called greedy strategy.

**Greedy strategy, based on exploitation**, is a simple strategy in which the agent always chooses the actions with the best rewards seen so far. In the MAB problem, it means that we always choose the slot machine that give us the best payout in a greedy way.

This strategy focuses on the idea of exploitation of the information gained from the agent’s experience so far. It may appear like a good strategy, but it has problems.

If we find one slot that seems to pay out well and stick with it, it may produce suboptimal results because we did not experiment with other slot machines, which may be better.

**Random strategy, based on exploration**, is an intuitive strategy where we randomly select a slot machine to play. Whereas the greedy strategy only chooses the action with the best payout up to that point, the random strategy is useful for exploring other possible actions.

Its problem is that does not exploit knowledge gained from the past rewards, to make more informed decisions over time.

We can see from the preceding two strategies that neither exploitation nor exploration alone are promising in the MAB problem. To address the shortcomings, can be employed a strategy of combining the two, called -greedy.

-greedy combines the exploitation and exploration. Epsilon is a hyperparameter, between 0 and 1, that determines whether a random selection is made to explore, or a greedy selection is made to exploit.

Usually, a random number between 0 and 1 is drawn. If that number is below Epsilon, we explore by selecting a random slot machine, otherwise we choose the machine that had the best payoff up to that point.

This helps us to continue to exploit the machine that provided best rewards so far, while still exploring other options.

Usually, a small value like 0.05 or 0.1 are attributed to Epsilon, in order for the agent to rely more on existing accumulated knowledge, than experimenting with new strategies , i.e. we want to exploit more than explore.

Although the Epsilon greedy strategy does not use random selection and it is more adaptive in the face of rime varying reward structures, it might still select an obviously suboptimal action in many random trials.

A refinement to the Epsilon greedy strategy is to allow Epsilon to vary systematically throughout the exercise, so that it is initially larger, allowing a lot of experimentation while the amount of accumulated knowledge about the relationships between actions and rewards is low.

Then, the hyperparameter is gradually reduced as more information becomes known and the benefit of additional exploration is diminished because the algorithm has already learned more about the test strategy.

A popular approach is to use a decay factor, , and set , where t is the trial number and with Beta between 0 and 1.

As an example, assume slot machine A,B and C and that the game can be played many times. The payoff from the ith machine is normally distributed with mean miu and Standard deviation of one. The means are known in advance.

And decay factor for Epsilon.

In the fist simulation, Epsilon equals one, as we know nothing about the slot machines and we must explore. We choose machine A and receive a payoff equal to 1.2. At this point, the value of Epsilon is now 0.85. Therefore, we drawn a random number between 0 and 1.

If the number below 0.85, we explore, vice versa we exploit.

Suppose that the Random number is 0.99, then we return to machine A to exploit. The payoff is now 0.8, so we will update our expected reward for machine A by averaging the two outcomes.

From the pictures below, although many other trials would be generally, needed, if we had to stop the process at this stage the best strategy identified would be to play slot machine A, because it was the highest payout ratio after 10 trials.

Also, as the number of trials increase, the use of exploration reduces thanks to de decay factor. The parameter Epsilon is close to 0 after about 57 trials, which implies that any new random number drawn will be greater than Epsilon and exploration stops. We juts paly the machine with the highest expected reward.

A screenshot of a white sheet with orange and blue text

Description automatically generated

A white grid with black text

Description automatically generated

In general, the new value of action-value fucntion, Qk, for the kth slot machine after it was chosen for n number of trials is:

Here Rj is the reward for the jth trial. It can be shown with simple algebra that each time slot machine K is chosen, its new Q-Value will be a weighted average of its old Q-value and the new reward, with weights and respectively.

The MAB problem is clearly much simpler than the setup for most potential applications of reinforcement learning because there is only on state here and the slot machines do not change. One example of problems with multiple states is the Markov decision process.

## **6.3 Markov Decision Processes**

Markov Decision Processes, MDPs, are simple settings for environment dynamics. In this case, the environment changes based on the actions of the agent. MDPs are processes that have no memory, which means that only the current state is relevant for determining the most appropriate current action and not any of the previous states.

MDPs are useful for modeling decision making in the cases where the agent is not fully in control of the evolution of the states. The use of MDPs establishes a straightforward framework where there are m states, denoted s, each of which will occur with a given probability, and there is also a fixed probability of being in a particular state at time t+1 given that the state at time t was .

The assumption that each state follows a Markov process greatly simplifies the analysis because such processes have no memory, as described earlier. We can express Markov property as:

That is, the future state at t+1 is only dependent on the current state at t, and independent of past states at t-1, t-2 etc. We can then specify a transition probability matrix, P, that shows the probabilities of moving from any state in one period to any other state in the next period.

The probabilities of being in each state n timesteps into the future, given an initial state, are then simply given by the elements P of n, i.e. the transition matrix P multiplied by itself n times.

The Markov assumption therefore provides a simple way for the algorithm to determine how likely each future state is given the current state.

The agent will select an action at time t, At, =a, based on observing state St=s, and receives a reward, Rt+1 in the next period at t+1 as a function of the state St+1=s’ in that period and the action in the previous period:

The agent goal will be to select the policy that maximizes this expected return aggregated over all feature time periods, note that the current and previous returns are “sunk costs” and hence are not included in the objective function).

Denoting the discount factor by , with Upsilon between zero and 1, we can define the goal at time t, Gt as:

The term in parentheses is the equation for Gt+1 and so the expression Gt could be further redrafted as a recursion:

Given the definition of Gt, we can define the state-value function V which measures how good a certain state s is following a certain Policy, Pi as :

We can also define the action-value function, Q, which measures how good a certain action a is, in a state s, following a certain policy, Pi, as its expected return:

Clearly, the agent’s objective will be to choose the optimal policy, Pi\*, that maximizes Q.

Where argmax is the set of values of Pi for which Qpi is maximized.

## **6.4 Approaches to Reinforcement Learning**

Reinforcement Learning algorithms can be classifies ad model-based and model-free algorithms. When there are a limited number of states, with well defined action and the transition probabilities are well defined, a dynamic programming technique can be used to obtain a solution.

Typically, however, we only have partial information about the model. In such cases, the algorithm search for an optimal solution to maximize the reward. There are two different approaches to find the policy that maximizes Q.

Value-based approaches that work on maximizing the reward by determining the best action in each state. Several algorithms such as Temporal difference method, Q-learning, SARSA and Deep Q-learning belong to this category.

Policy-based approaches that find the optimal policy to map a state into an action. The Policy gradient approach is a commonly used policy-based algorithm.

## **Appendix A -Q&A Module 1 from CHAT GPT**

**1. Who is considered the brainchild behind the concept of the digital computer?**

A) Charles Babbage  
B) John McCarthy  
C) Alan Turing  
D) Warren McCulloch  
E) Donald Hebb  
F) Herbert Simon

**2. What concept did Alan Turing introduce in his 1950 paper "Computing Machinery and Intelligence"?**

A) Neural Networks  
B) Machine Learning  
C) Reinforcement Learning  
D) The Imitation Game (Turing Test)  
E) Expert Systems  
F) Search Algorithms

**3. Which of the following is an example of a classical AI technique?**

A) Deep Learning  
B) Connectionism  
C) Symbolic Logic  
D) Unsupervised Learning  
E) Neural Networks  
F) Reinforcement Learning

**4. What is the main challenge associated with classical AI when dealing with real-world scenarios?**

A) Lack of computing power  
B) Inability to mimic human reasoning  
C) Difficulty in representing characteristics and relationships of real-world entities  
D) Complexity of neural networks  
E) Over-reliance on expert systems  
F) Lack of training data

**5. What is the "Frame Problem" in classical AI?**

A) The challenge of designing AI to frame pictures  
B) Keeping track of which aspects of a situation change and which stay the same after an action is performed  
C) Difficulties in creating frames for data representation  
D) A method of representing problem spaces in AI  
E) A limitation in the memory capacity of early AI systems  
F) The problem of ensuring AI operates within a specific frame of reference

**6. Which search technique uses a heuristic to estimate the best path to take?**

A) Binary Search  
B) Depth-First Search  
C) Breadth-First Search  
D) A\* Algorithm  
E) Adversarial Search  
F) Recursive Search

**7. What role does reinforcement learning play in contemporary AI?**

A) It allows AI to play games against itself  
B) It enables AI to learn from mistakes and improve performance  
C) It is used to develop neural networks  
D) It helps AI systems make decisions based on logic  
E) It automates the reasoning process  
F) It helps create expert systems

**8. What is a recursive function?**

A) A function that calls itself  
B) A function that uses neural networks  
C) A function that searches through all possibilities  
D) A function that simplifies data  
E) A function that automates reasoning  
F) A function that predicts outcomes

**9. Which game is often used as an illustration of recursive techniques in AI?**

A) Chess  
B) Checkers  
C) Tower of Hanoi  
D) Tic-Tac-Toe  
E) Go  
F) Sudoku

**10. Which concept in AI is concerned with reducing the complexity of a task by dividing it into simpler tasks of the same type?**

A) Heuristic Search  
B) Reinforcement Learning  
C) Adversarial Search  
D) Recursion  
E) Binary Search  
F) Neural Network

**11. What challenge did early AI researchers encounter when applying general-purpose search mechanisms?**

A) Insufficient computational power  
B) Combinatorial explosion  
C) Lack of accurate data  
D) Difficulty in coding logic  
E) Over-reliance on heuristics  
F) The problem of recursion

**12. What was the limitation of expert systems in classical AI?**

A) They could not be used in real-world scenarios  
B) They were too slow  
C) They were difficult to program  
D) They could not be easily generalized beyond specific domains  
E) They were not reliable  
F) They could not process natural language

**13. What does a neural network attempt to mimic?**

A) The logical reasoning of a human  
B) The decision-making process of an expert system  
C) The behavior of neurons in the brain  
D) The search techniques of classical AI  
E) The structure of data  
F) The rules of a game

**14. Who contributed to the development of the "neurophysiological postulate" in 1949?**

A) Alan Turing  
B) John McCarthy  
C) Donald Hebb  
D) Warren McCulloch  
E) Walter Pitts  
F) Herbert Simon

**15. What is the main characteristic of a "smart" or "intelligent" system according to classical AI?**

A) Ability to learn autonomously  
B) Ability to solve specific problems  
C) Ability to mimic human emotions  
D) General intelligence  
E) Neural connectivity  
F) Ability to generate natural language

**16. What does the A\* algorithm prioritize when searching for a solution?**

A) The longest path  
B) The path with the least number of steps  
C) The most promising path based on heuristic estimates  
D) The first path it finds  
E) The path with the most rewards  
F) The path with the highest risk

**17. Which type of learning is based on observing outcomes and reinforcing successful actions?**

A) Supervised Learning  
B) Unsupervised Learning  
C) Reinforcement Learning  
D) Deep Learning  
E) Connectionist Learning  
F) Symbolic Learning

**18. What issue arises from the "curse of exponential complexity" in AI?**

A) AI systems become too complex to function  
B) The number of possibilities to analyze becomes too large  
C) AI fails to understand simple tasks  
D) Heuristics become ineffective  
E) AI cannot process large datasets  
F) The search space becomes too small

**19. What is a key advantage of using heuristic search techniques in AI?**

A) They guarantee an optimal solution  
B) They reduce the search space  
C) They eliminate the need for human intervention  
D) They make AI systems more reliable  
E) They improve the accuracy of predictions  
F) They simplify the coding process

**20. What is one of the risks associated with over-reliance on AI systems?**

A) Decreased computational power  
B) Overly complex algorithms  
C) Inability to adapt to new situations  
D) Loss of human expertise  
E) Difficulty in coding  
F) Increase in computational errors

**21. What is the primary purpose of the Turing Test in AI?**

A) To test AI's problem-solving abilities  
B) To evaluate the efficiency of AI algorithms  
C) To determine whether a machine can exhibit human-like intelligence  
D) To assess the computational power of AI  
E) To evaluate AI's ability to learn from data  
F) To measure the speed of AI processing

**22. Which type of AI is focused on performing specific tasks with high proficiency?**

A) General AI  
B) Weak AI  
C) Strong AI  
D) Conscious AI  
E) Autonomous AI  
F) Ethical AI

**23. What is a common limitation of neural networks in AI?**

A) Inability to process natural language  
B) Lack of scalability  
C) Difficulty in interpreting their decision-making process  
D) Slow processing speed  
E) High error rates  
F) Limited application in real-world scenarios

**24. What term is used to describe AI systems that can improve their performance over time without human intervention?**

A) Adaptive AI  
B) Heuristic AI  
C) Static AI  
D) Rule-Based AI  
E) Dynamic AI  
F) Reinforcement AI

**25. Which AI concept involves the machine's ability to understand and generate human language?**

A) Neural Networks  
B) Symbolic AI  
C) Natural Language Processing (NLP)  
D) Deep Learning  
E) Reinforcement Learning  
F) Expert Systems

**26. What is the goal of supervised learning in AI?**

A) To classify data into predefined categories  
B) To detect patterns in unlabelled data  
C) To enhance reinforcement learning  
D) To mimic human decision-making  
E) To optimize search algorithms  
F) To automate the reasoning process

**27. Which AI technique is most commonly used in recommendation systems?**

A) Genetic Algorithms  
B) Symbolic Logic  
C) Collaborative Filtering  
D) Adversarial Search  
E) Reinforcement Learning  
F) Decision Trees

**28. What is the function of an activation function in a neural network?**

A) To initialize the network's weights  
B) To control the output of a neuron  
C) To optimize the network's performance  
D) To determine the learning rate  
E) To preprocess the input data  
F) To minimize the loss function

**29. Which method is used to prevent overfitting in machine learning models?**

A) Increasing the size of the dataset  
B) Decreasing the learning rate  
C) Reducing the number of layers in the model  
D) Regularization  
E) Using a smaller batch size  
F) Applying a higher dropout rate

**30. What is the main challenge in unsupervised learning?**

A) Lack of labeled data  
B) High computational cost  
C) Inability to handle large datasets  
D) Complexity in model interpretation  
E) Limited generalization ability  
F) Difficulty in measuring model performance

**31. Which AI paradigm focuses on simulating the human brain's neural structure?**

A) Symbolic AI  
B) Expert Systems  
C) Neural Networks  
D) Genetic Algorithms  
E) Logic Programming  
F) Case-Based Reasoning

**32. What is the role of a loss function in machine learning?**

A) To evaluate the performance of a model  
B) To optimize the network's architecture  
C) To preprocess the input data  
D) To control the activation of neurons  
E) To determine the model's hyperparameters  
F) To initialize the network's weights

**33. In AI, what does "overfitting" refer to?**

A) A model that performs well on training data but poorly on new data  
B) A model that is too complex for the problem it is solving  
C) A model that underestimates the complexity of the data  
D) A model that has too many parameters  
E) A model that requires too much computational power  
F) A model that cannot learn from new data

**34. Which AI technique is inspired by the process of natural selection?**

A) Neural Networks  
B) Genetic Algorithms  
C) Reinforcement Learning  
D) Symbolic Logic  
E) Deep Learning  
F) Expert Systems

**35. What is the primary advantage of deep learning over traditional machine learning?**

A) Faster computation  
B) Ability to process unstructured data  
C) Simpler algorithms  
D) Lower computational cost  
E) Higher accuracy on small datasets  
F) Easier model interpretation

**36. Which AI model is commonly used in image recognition tasks?**

A) Recurrent Neural Networks (RNNs)  
B) Decision Trees  
C) Convolutional Neural Networks (CNNs)  
D) Support Vector Machines (SVMs)  
E) K-Nearest Neighbors (KNN)  
F) Naive Bayes Classifier

**37. What does the term "big data" refer to in the context of AI?**

A) Extremely large datasets that cannot be processed by traditional methods  
B) The use of deep learning models on large datasets  
C) The application of AI to real-world problems  
D) Data generated by neural networks  
E) Data that is highly structured and organized  
F) Data that is collected from a variety of sources

**38. Which AI technique involves the use of "agents" to solve complex problems?**

A) Deep Learning  
B) Symbolic AI  
C) Multi-Agent Systems  
D) Genetic Algorithms  
E) Neural Networks  
F) Case-Based Reasoning

**39. What is the purpose of backpropagation in neural networks?**

A) To forward the input through the network  
B) To initialize the weights of the network  
C) To compute the error in the output  
D) To update the weights of the network based on the error  
E) To optimize the activation functions  
F) To determine the structure of the network

**40. Which of the following is an example of unsupervised learning?**

A) Decision Trees  
B) K-Means Clustering  
C) Random Forests  
D) Support Vector Machines  
E) Reinforcement Learning  
F) Logistic Regression

**41. What is the primary goal of AI ethics?**

A) To create AI systems that are unbiased  
B) To enhance the efficiency of AI algorithms  
C) To ensure AI systems operate within legal frameworks  
D) To prevent AI from making mistakes  
E) To understand the moral implications of AI  
F) To increase AI's decision-making capabilities

**42. Which type of AI can make decisions without human intervention?**

A) Weak AI  
B) Strong AI  
C) Autonomous AI  
D) Collaborative AI  
E) Ethical AI  
F) Symbolic AI

**43. What is the main characteristic of a convolutional neural network (CNN)?**

A) It uses feedback loops for prediction  
B) It processes data sequentially  
C) It is primarily used for natural language processing  
D) It is designed to recognize patterns in images  
E) It relies on symbolic logic  
F) It simulates the decision-making process

**44. What challenge does the "black box" problem in AI refer to?**

A) The difficulty in understanding and interpreting the decision-making process of AI models  
B) The challenge of storing large datasets  
C) The inability of AI to learn from new data  
D) The problem of AI systems being too slow  
E) The risk of AI systems becoming too complex  
F) The issue of AI models being too simple

**45. What is the role of a training dataset in machine learning?**

A) To test the model's performance  
B) To fine-tune the model's hyperparameters  
C) To validate the model's predictions  
D) To learn the model's parameters  
E) To preprocess the data before modeling  
F) To evaluate the model's accuracy

**46. Which of the following is a type of recurrent neural network (RNN) architecture?**

A) Long Short-Term Memory (LSTM)  
B) Convolutional Neural Network (CNN)  
C) Decision Tree  
D) Support Vector Machine (SVM)  
E) Naive Bayes  
F) K-Nearest Neighbors (KNN)

**47. What does "scalability" refer to in the context of AI systems?**

A) The ability to handle increasing amounts of data or computational resources  
B) The ability to be implemented in different environments  
C) The ability to be easily understood by humans  
D) The ability to solve a wide range of problems  
E) The ability to reduce computational costs  
F) The ability to generate new data

**48. What is one of the primary concerns related to AI and job displacement?**

A) AI systems will require too much human oversight  
B) AI will replace jobs faster than new jobs can be created  
C) AI systems will make too many errors  
D) AI will create jobs that are too specialized  
E) AI will not be able to perform tasks as well as humans  
F) AI systems will be too expensive to implement

**49. Which AI technique involves creating a model that can predict outcomes based on input data?**

A) Reinforcement Learning  
B) Unsupervised Learning  
C) Supervised Learning  
D) Genetic Algorithms  
E) Expert Systems  
F) Case-Based Reasoning

**50. What is the goal of transfer learning in AI?**

A) To use knowledge gained from one task to improve performance on another  
B) To transfer data between different AI models  
C) To optimize the performance of neural networks  
D) To enhance the learning rate of AI models  
E) To create models that can be easily interpreted  
F) To develop AI systems that can operate in real-time

**Answers:**

1. C) Alan Turing
2. D) The Imitation Game (Turing Test)
3. C) Symbolic Logic
4. C) Difficulty in representing characteristics and relationships of real-world entities
5. B) Keeping track of which aspects of a situation change and which stay the same after an action is performed
6. D) A\* Algorithm
7. B) It enables AI to learn from mistakes and improve performance
8. A) A function that calls itself
9. C) Tower of Hanoi
10. D) Recursion
11. B) Combinatorial explosion
12. D) They could not be easily generalized beyond specific domains
13. C) The behavior of neurons in the brain
14. C) Donald Hebb
15. B) Ability to solve specific problems
16. C) The most promising path based on heuristic estimates
17. C) Reinforcement Learning
18. B) The number of possibilities to analyze becomes too large
19. B) They reduce the search space
20. D) Loss of human expertise
21. C) To determine whether a machine can exhibit human-like intelligence
22. B) Weak AI
23. C) Difficulty in interpreting their decision-making process
24. A) Adaptive AI
25. C) Natural Language Processing (NLP)
26. A) To classify data into predefined categories
27. C) Collaborative Filtering
28. B) To control the output of a neuron
29. D) Regularization
30. A) Lack of labeled data
31. C) Neural Networks
32. A) To evaluate the performance of a model
33. A) A model that performs well on training data but poorly on new data
34. B) Genetic Algorithms
35. B) Ability to process unstructured data
36. C) Convolutional Neural Networks (CNNs)
37. A) Extremely large datasets that cannot be processed by traditional methods
38. C) Multi-Agent Systems
39. D) To update the weights of the network based on the error
40. B) K-Means Clustering
41. E) To understand the moral implications of AI
42. C) Autonomous AI
43. D) It is designed to recognize patterns in images
44. A) The difficulty in understanding and interpreting the decision-making process of AI models
45. D) To learn the model's parameters
46. A) Long Short-Term Memory (LSTM)
47. A) The ability to handle increasing amounts of data or computational resources
48. B) AI will replace jobs faster than new jobs can be created
49. C) Supervised Learning
50. A) To use knowledge gained from one task to improve performance on another

## **Appendix B -Q&A Module 2 from CHAT GPT**

1. **What is the primary advantage of Machine Learning over traditional econometrics?**
   * A. Handling big data
   * B. Simpler algorithms
   * C. Linear modeling
   * D. Pre-specified theories
   * E. R-squared value optimization
   * F. No need for data cleaning
2. **Which Machine Learning technique is used to reduce dimensionality?**
   * A. Neural networks
   * B. Regression analysis
   * C. Principal Component Analysis (PCA)
   * D. Decision trees
   * E. Clustering
   * F. K-nearest neighbors
3. **Which of the following is NOT a characteristic of Machine Learning?**
   * A. Out-of-sample prediction
   * B. Pre-specified model theories
   * C. Handling missing data
   * D. Model selection
   * E. Regularization techniques
   * F. Dimensionality reduction
4. **Which type of Machine Learning is primarily concerned with prediction and classification?**
   * A. Supervised Learning
   * B. Unsupervised Learning
   * C. Reinforcement Learning
   * D. Semi-supervised Learning
   * E. Parametric Learning
   * F. Nonparametric Learning
5. **Which of the following is a key component of Unsupervised Learning?**
   * A. Prediction accuracy
   * B. Labeling data
   * C. Dimensionality reduction
   * D. Parameter estimation
   * E. Model inference
   * F. Outlier detection
6. **Which type of Machine Learning involves feedback provided in the form of rewards?**
   * A. Supervised Learning
   * B. Unsupervised Learning
   * C. Semi-supervised Learning
   * D. Reinforcement Learning
   * E. Parametric Learning
   * F. Nonparametric Learning
7. **What is the role of Exploratory Data Analysis (EDA) in Machine Learning?**
   * A. Model prediction
   * B. Parameter tuning
   * C. Data collection and cleaning
   * D. Model inference
   * E. Outlier removal
   * F. Algorithm selection
8. **Which of the following is an example of structured data?**
   * A. Web logs
   * B. Sensor data
   * C. Census data
   * D. Images
   * E. Network traffic
   * F. Texts
9. **What type of data refers to attributes like age and income?**
   * A. Categorical data
   * B. Binary data
   * C. Nominal data
   * D. Ordinal data
   * E. Numerical data
   * F. Textual data
10. **Which of the following data types is concerned with attributes that have no natural ordering?**
    * A. Interval data
    * B. Ratio data
    * C. Ordinal data
    * D. Nominal data
    * E. Binary data
    * F. Categorical data
11. **In which type of data are observations related temporally, spatially, or through network connections?**
    * A. Cross-sectional data
    * B. Longitudinal data
    * C. Nominal data
    * D. Ordinal data
    * E. Interval data
    * F. Ratio data
12. **Which Machine Learning process involves transforming non-numeric information into numbers?**
    * A. Normalization
    * B. Standardization
    * C. Data scaling
    * D. Encoding
    * E. Feature extraction
    * F. Imputation
13. **What is the term used for removing unwanted observations from a dataset?**
    * A. Imputation
    * B. Data cleaning
    * C. Outlier detection
    * D. Data visualization
    * E. Standardization
    * F. Normalization
14. **What is a common method used to handle missing data in large datasets?**
    * A. Outlier removal
    * B. K-nearest neighbors
    * C. Standardization
    * D. Normalization
    * E. Principal Component Analysis
    * F. Regression analysis
15. **Which scale of measurement has a true zero point?**
    * A. Nominal
    * B. Ordinal
    * C. Interval
    * D. Ratio
    * E. Binary
    * F. Continuous
16. **Which process creates a new variable with a mean of zero and variance of one?**
    * A. Normalization
    * B. Standardization
    * C. Data cleaning
    * D. Encoding
    * E. Feature extraction
    * F. Imputation
17. **Which of the following transformations is commonly used when data is highly skewed?**
    * A. Standardization
    * B. Normalization
    * C. Log transformation
    * D. Encoding
    * E. Data scaling
    * F. Imputation
18. **Which technique is most appropriate when dealing with highly correlated variables in a dataset?**
    * A. Feature extraction
    * B. Principal Component Analysis (PCA)
    * C. Data scaling
    * D. Data cleaning
    * E. Encoding
    * F. Imputation
19. **What is the primary purpose of Principal Component Analysis (PCA)?**
    * A. Model prediction
    * B. Data cleaning
    * C. Dimensionality reduction
    * D. Feature extraction
    * E. Data visualization
    * F. Model inference
20. **Which method is typically used to address multicollinearity in a dataset?**
    * A. Normalization
    * B. Standardization
    * C. Data cleaning
    * D. Principal Component Analysis (PCA)
    * E. Feature extraction
    * F. Imputation
21. **Which Machine Learning method does NOT require assumptions about the functional form of the relationship between features and labels?**
    * A. Parametric methods
    * B. Nonparametric methods
    * C. Supervised learning
    * D. Unsupervised learning
    * E. Semi-supervised learning
    * F. Reinforcement learning
22. **What does the acronym GIGO stand for in the context of data analysis?**
    * A. Garbage In, Garbage Out
    * B. General Input, General Output
    * C. Good Input, Good Output
    * D. Generalized Inference, Generalized Output
    * E. Guided Inference, Guided Output
    * F. Gradual Input, Gradual Output
23. **What is the main challenge when dealing with unstructured data?**
    * A. Lack of dimensionality
    * B. Difficulty in visualization
    * C. Complexity in representation
    * D. Handling missing values
    * E. Multicollinearity
    * F. Data cleaning
24. **Which of the following is a type of semi-structured data?**
    * A. Texts
    * B. Census data
    * C. Geo-satellite images
    * D. Web logs
    * E. Sensor data
    * F. Images
25. **What is the term for variables that have a meaningful order but no clear interval between values?**
    * A. Nominal data
    * B. Ordinal data
    * C. Interval data
    * D. Ratio data
    * E. Binary data
    * F. Continuous data
26. **Which process is necessary to avoid multicollinearity when using dummy variables in regression models?**
    * A. Data scaling
    * B. Normalization
    * C. Standardization
    * D. Encoding
    * E. Excluding one dummy variable
    * F. Imputation
27. **Which scale of measurement is appropriate for temperature readings?**
    * A. Nominal
    * B. Ordinal
    * C. Interval
    * D. Ratio
    * E. Binary
    * F. Continuous
28. **Which process is used to convert numeric attributes into categorical ones?**
    * A. Standardization
    * B. Normalization
    * C. Discretization
    * D. Encoding
    * E. Feature extraction
    * F. Data cleaning
29. **Which process is most appropriate for handling outliers when the model is sensitive to them?**
    * A. Data scaling
    * B. Normalization
    * C. Standardization
    * D. Encoding
    * E. Feature extraction
    * F. Imputation
30. **What technique is commonly used to visualize the distribution of a single variable?**
    * A. Scatter plot
    * B. Line chart
    * C. Histogram
    * D. Box plot
    * E. Bar chart
    * F. Pie chart
31. **Which type of variable has a consistent interval between values but no true zero point?**
    * A. Nominal data
    * B. Ordinal data
    * C. Interval data
    * D. Ratio data
    * E. Binary data
    * F. Continuous data
32. **What is the key difference between parametric and nonparametric methods?**
    * A. Parametric methods are more flexible
    * B. Nonparametric methods require more assumptions
    * C. Parametric methods require assumptions about the data distribution
    * D. Nonparametric methods work only with small datasets
    * E. Parametric methods are less accurate
    * F. Nonparametric methods require less data cleaning
33. **Which of the following is NOT a data transformation technique?**
    * A. Normalization
    * B. Standardization
    * C. Imputation
    * D. Encoding
    * E. Feature extraction
    * F. Discretization
34. **Which term refers to data that is organized and stored in a predefined format?**
    * A. Unstructured data
    * B. Semi-structured data
    * C. Structured data
    * D. Temporal data
    * E. Spatial data
    * F. Network data
35. **Which Machine Learning method is best suited for identifying relationships in large datasets?**
    * A. Supervised learning
    * B. Unsupervised learning
    * C. Semi-supervised learning
    * D. Reinforcement learning
    * E. Nonparametric learning
    * F. Parametric learning
36. **What is the primary goal of data normalization?**
    * A. Reducing data redundancy
    * B. Removing noise from data
    * C. Ensuring data integrity
    * D. Adjusting data to a common scale
    * E. Removing outliers
    * F. Encoding categorical data
37. **Which method would you use to visualize the relationship between two continuous variables?**
    * A. Histogram
    * B. Pie chart
    * C. Line chart
    * D. Scatter plot
    * E. Box plot
    * F. Bar chart
38. **Which of the following is typically done during data preprocessing?**
    * A. Model selection
    * B. Model evaluation
    * C. Data cleaning
    * D. Model inference
    * E. Parameter tuning
    * F. Algorithm optimization
39. **Which of the following is a key advantage of using PCA?**
    * A. Increases the number of features
    * B. Decreases the accuracy of predictions
    * C. Reduces the dimensionality of the dataset
    * D. Makes data interpretation more difficult
    * E. Requires more computational resources
    * F. Removes all noise from the data
40. **Which transformation technique is most effective when dealing with skewed data?**
    * A. Log transformation
    * B. Standardization
    * C. Normalization
    * D. Encoding
    * E. Data scaling
    * F. Imputation
41. **Which of the following is an example of unstructured data?**
    * A. Census data
    * B. Survey responses
    * C. Sensor readings
    * D. Social media posts
    * E. Financial transactions
    * F. Time series data
42. **Which method is typically used to fill in missing data by taking the most frequent value?**
    * A. Median imputation
    * B. Mean imputation
    * C. Mode imputation
    * D. Forward filling
    * E. Backward filling
    * F. Random imputation
43. **What is the goal of feature extraction?**
    * A. Increase the number of variables
    * B. Reduce the number of variables
    * C. Simplify data cleaning
    * D. Improve data scaling
    * E. Enhance data visualization
    * F. Improve model interpretability
44. **Which of the following is NOT an example of continuous data?**
    * A. Height
    * B. Weight
    * C. Age
    * D. Gender
    * E. Temperature
    * F. Distance
45. **Which data type is best visualized with a bar chart?**
    * A. Nominal data
    * B. Ordinal data
    * C. Interval data
    * D. Ratio data
    * E. Binary data
    * F. Continuous data
46. **Which process helps to address the problem of multicollinearity?**
    * A. Data scaling
    * B. Normalization
    * C. Principal Component Analysis (PCA)
    * D. Standardization
    * E. Data cleaning
    * F. Encoding
47. **Which of the following is an example of categorical data?**
    * A. Temperature
    * B. Distance
    * C. Gender
    * D. Age
    * E. Income
    * F. Weight
48. **Which of the following is NOT a method of data imputation?**
    * A. Mean imputation
    * B. Mode imputation
    * C. Median imputation
    * D. Forward filling
    * E. Backward filling
    * F. Data scaling
49. **Which of the following is a common measure to handle multicollinearity in regression analysis?**
    * A. Standardization
    * B. Normalization
    * C. Dimensionality reduction
    * D. Excluding correlated variables
    * E. Feature extraction
    * F. Encoding
50. **Which visualization method is used to show the distribution of categorical data?**
    * A. Scatter plot
    * B. Histogram
    * C. Line chart
    * D. Box plot
    * E. Pie chart
    * F. Bar chart

**Answers:**

1. A
2. C
3. B
4. A
5. C
6. D
7. C
8. C
9. E
10. D
11. B
12. D
13. B
14. B
15. D
16. B
17. C
18. B
19. C
20. D
21. B
22. A
23. C
24. D
25. B
26. E
27. C
28. C
29. F
30. C
31. C
32. C
33. C
34. C
35. B
36. D
37. D
38. C
39. C
40. A
41. D
42. C
43. B
44. D
45. A
46. C
47. C
48. F
49. D
50. F

## **Appendix B -Q&A Module 2 Chapter 2 from CHAT GPT**

1. **What is the main goal of unsupervised learning?**
   * A) Predicting future outcomes
   * B) Supervising labeled data
   * C) Developing insights or pattern recognition with no specific guidance
   * D) Validating models
   * E) Enhancing data visualization
   * F) Solving regression problems
2. **Which technique is commonly associated with unsupervised learning?**
   * A) Decision Trees
   * B) Clustering Analysis
   * C) Linear Regression
   * D) Support Vector Machines
   * E) Gradient Boosting
   * F) Logistic Regression
3. **Clustering places similar data points into the same group based on what?**
   * A) Time
   * B) Distance
   * C) Weight
   * D) Age
   * E) Color
   * F) Shape
4. **Which of the following is NOT a clustering technique?**
   * A) K-Means
   * B) DBSCAN
   * C) Hierarchical Clustering
   * D) Random Forest
   * E) Agglomerative Clustering
   * F) Divisive Clustering
5. **What is essential for data before performing clustering analysis?**
   * A) Normalization or Standardization
   * B) Deletion of outliers
   * C) Labeling
   * D) Feature engineering
   * E) Model training
   * F) Cross-validation
6. **What type of clustering begins with each data point in a separate cluster and successively combines them?**
   * A) Partitional Clustering
   * B) Divisive Clustering
   * C) Density-Based Clustering
   * D) Agglomerative Clustering
   * E) K-Means Clustering
   * F) Spectral Clustering
7. **What is a dendrogram?**
   * A) A type of decision tree
   * B) A hierarchical clustering technique
   * C) A machine learning algorithm
   * D) A neural network
   * E) A clustering validation method
   * F) A regression analysis tool
8. **Which algorithm is an example of partitional clustering?**
   * A) K-Means
   * B) DBSCAN
   * C) Agglomerative Clustering
   * D) Divisive Clustering
   * E) Spectral Clustering
   * F) Ward's Method
9. **Inertia in clustering measures what?**
   * A) The distance between data points
   * B) The within-cluster sum of squares
   * C) The silhouette score
   * D) The correlation between features
   * E) The variance of the data
   * F) The density of the data points
10. **Which score compares within-cluster distances and distances between clusters?**
    * A) Inertia
    * B) Silhouette Score
    * C) Variance Ratio Criterion
    * D) Cross-Validation Score
    * E) Mean Squared Error
    * F) F1 Score
11. **Which of the following is NOT an issue with K-Means clustering?**
    * A) Non-spherical clusters
    * B) Presence of outliers
    * C) Curse of dimensionality
    * D) Necessity for labeled data
    * E) Sensitivity to initial centroid positions
    * F) Poor performance with high-dimensional data
12. **What is the primary method used to calculate distances in K-Means?**
    * A) Manhattan Distance
    * B) Hamming Distance
    * C) Euclidean Distance
    * D) Mahalanobis Distance
    * E) Cosine Similarity
    * F) Minkowski Distance
13. **Which of the following is an advantage of the K-Means algorithm?**
    * A) Handles non-spherical clusters effectively
    * B) Always finds the global minimum
    * C) Fast and scales well to large datasets
    * D) Requires fewer iterations
    * E) Robust to outliers
    * F) Does not require specifying the number of clusters
14. **Which clustering technique does not require the number of clusters to be specified a priori?**
    * A) K-Means
    * B) DBSCAN
    * C) Spectral Clustering
    * D) Agglomerative Clustering
    * E) Partitional Clustering
    * F) Divisive Clustering
15. **What distance measure is considered the multivariate equivalent of the Z-Score?**
    * A) Euclidean Distance
    * B) Manhattan Distance
    * C) Minkowski Distance
    * D) Mahalanobis Distance
    * E) Cosine Similarity
    * F) Chebyshev Distance
16. **Which clustering algorithm uses density points to form clusters?**
    * A) K-Means
    * B) Spectral Clustering
    * C) DBSCAN
    * D) Agglomerative Clustering
    * E) Partitional Clustering
    * F) Divisive Clustering
17. **Which of the following statements about the Silhouette Score is true?**
    * A) It ranges from 0 to 1
    * B) It is used to assess linearity
    * C) A higher score implies better clustering
    * D) It measures the sum of squared errors
    * E) It compares model predictions with actual outcomes
    * F) It indicates the number of outliers in a dataset
18. **Which measure would best handle non-spherical clusters?**
    * A) Inertia
    * B) Silhouette Score
    * C) Manhattan Distance
    * D) Cosine Similarity
    * E) Density-based clustering methods
    * F) Mahalanobis Distance
19. **What does the K in K-Means represent?**
    * A) The total number of clusters
    * B) The number of centroids
    * C) The initial seed value
    * D) The distance metric used
    * E) The number of iterations
    * F) The kernel used in the model
20. **Which technique adjusts initial centroid positions to be far from each other?**
    * A) Random Initialization
    * B) K-Means++
    * C) PCA Initialization
    * D) Hierarchical Clustering
    * E) Gaussian Mixture Models
    * F) Agglomerative Clustering
21. **What is a common issue with randomly selected centroids in K-Means?**
    * A) All points are equally distant from the centroid
    * B) Centroids may not be distinct enough
    * C) It always leads to the global minimum
    * D) The process is deterministic
    * E) No convergence is possible
    * F) Clusters are always spherical
22. **Which method helps determine the optimal number of clusters in K-Means?**
    * A) Elbow Method
    * B) Inertia Plot
    * C) Scree Plot
    * D) Variance Ratio Criterion
    * E) Cross-Validation
    * F) Decision Trees
23. **Which algorithm does not require rescaling of data for clustering?**
    * A) K-Means
    * B) DBSCAN
    * C) Agglomerative Clustering
    * D) Hierarchical Clustering
    * E) K-Means++
    * F) Divisive Clustering
24. **In density-based clustering, what is a 'core point'?**
    * A) A point within the highest density region
    * B) The most central point in a cluster
    * C) A point that exceeds the pre-specified density threshold
    * D) The furthest point from the centroid
    * E) A point that is a clear outlier
    * F) A point with the smallest Euclidean distance
25. **Which of the following is a non-parametric clustering algorithm?**
    * A) K-Means
    * B) Hierarchical Clustering
    * C) DBSCAN
    * D) Spectral Clustering
    * E) Gaussian Mixture Model
    * F) Divisive Clustering
26. **What is the Manhattan distance used for?**
    * A) Measuring the distance along grid-like paths
    * B) Calculating Euclidean distance
    * C) Determining the most probable centroid
    * D) Clustering spherical data
    * E) Identifying outliers
    * F) Assessing model accuracy
27. **Which clustering method would be least affected by outliers?**
    * A) K-Means
    * B) DBSCAN
    * C) Spectral Clustering
    * D) Agglomerative Clustering
    * E) K-Means++
    * F) Divisive Clustering
28. **Which distance metric would be most appropriate for high-dimensional data?**
    * A) Euclidean Distance
    * B) Manhattan Distance
    * C) Minkowski Distance
    * D) Cosine Similarity
    * E) Mahalanobis Distance
    * F) Chebyshev Distance
29. **Which method is an example of a linkage criterion used in hierarchical clustering?**
    * A) Single Linkage
    * B) Inertia
    * C) Euclidean Distance
    * D) K-Means++
    * E) Mahalanobis Distance
    * F) Silhouette Score
30. **Which algorithm is an extension of K-Means that helps to mitigate issues with random initialization?**
    * A) Random Initialization
    * B) K-Means++
    * C) Hierarchical Clustering
    * D) Spectral Clustering
    * E) DBSCAN
    * F) Gaussian Mixture Models
31. **What is the primary focus of spectral clustering?**
    * A) Optimizing centroids
    * B) Utilizing graph theory for clustering
    * C) Handling density-based clustering
    * D) Minimizing sum of squares
    * E) Maximizing distance between clusters
    * F) Creating a dendrogram
32. **Which clustering method forms a tree-like structure?**
    * A) Hierarchical Clustering
    * B) K-Means
    * C) Spectral Clustering
    * D) DBSCAN
    * E) Gaussian Mixture Model
    * F) Random Forest
33. **What is a disadvantage of hierarchical clustering?**
    * A) The number of clusters must be pre-specified
    * B) It is highly sensitive to outliers
    * C) It struggles with non-spherical data
    * D) The clustering structure is irreversible
    * E) It requires high-dimensional data
    * F) It uses Euclidean distance exclusively
34. **What kind of data does DBSCAN perform poorly on?**
    * A) Low-dimensional data
    * B) Data with uniform density
    * C) Spherical data
    * D) High-density data
    * E) Noisy data
    * F) Multivariate data
35. **Which criterion is often used to merge clusters in agglomerative clustering?**
    * A) Centroid distance
    * B) Average linkage
    * C) Variance Ratio Criterion
    * D) Silhouette Score
    * E) Euclidean distance
    * F) Inertia
36. **Which clustering method is preferred for detecting arbitrarily shaped clusters?**
    * A) K-Means
    * B) DBSCAN
    * C) Hierarchical Clustering
    * D) Spectral Clustering
    * E) Gaussian Mixture Model
    * F) Divisive Clustering
37. **What is the function of a 'reachability distance' in DBSCAN?**
    * A) Determining the distance between centroids
    * B) Assessing the connectivity of core points
    * C) Calculating the distance between all points
    * D) Identifying outliers
    * E) Defining the maximum distance within a cluster
    * F) Measuring the distance to the nearest cluster boundary
38. **Which clustering method operates by dividing the dataset into smaller clusters?**
    * A) Divisive Clustering
    * B) DBSCAN
    * C) K-Means
    * D) Agglomerative Clustering
    * E) Spectral Clustering
    * F) Gaussian Mixture Model
39. **What does a low Silhouette Score indicate?**
    * A) Well-separated clusters
    * B) Dense clusters
    * C) Overlapping clusters
    * D) Perfect clustering
    * E) High inertia
    * F) Few data points
40. **Which of the following metrics is NOT typically used in clustering validation?**
    * A) Silhouette Score
    * B) Inertia
    * C) F1 Score
    * D) Dunn Index
    * E) Rand Index
    * F) Davies-Bouldin Index
41. **What is a primary drawback of density-based clustering?**
    * A) It requires the number of clusters to be specified
    * B) It does not perform well on datasets with varying density
    * C) It is computationally expensive
    * D) It always finds spherical clusters
    * E) It cannot handle high-dimensional data
    * F) It relies heavily on the distance metric
42. **Which method is specifically designed to handle overlapping clusters?**
    * A) K-Means
    * B) DBSCAN
    * C) Spectral Clustering
    * D) Gaussian Mixture Models
    * E) Hierarchical Clustering
    * F) Divisive Clustering
43. **Which of the following statements about DBSCAN is correct?**
    * A) It is sensitive to noise
    * B) It requires a predefined number of clusters
    * C) It cannot handle large datasets
    * D) It is efficient in finding arbitrarily shaped clusters
    * E) It is based on a tree structure
    * F) It assumes Gaussian distribution
44. **Which method is a probabilistic model that assigns a probability to each point belonging to a cluster?**
    * A) DBSCAN
    * B) K-Means
    * C) Spectral Clustering
    * D) Gaussian Mixture Models
    * E) Agglomerative Clustering
    * F) Divisive Clustering
45. **Which clustering algorithm is known to be deterministic?**
    * A) DBSCAN
    * B) K-Means
    * C) Spectral Clustering
    * D) Agglomerative Clustering
    * E) Gaussian Mixture Models
    * F) Hierarchical Clustering
46. **Which criterion does Ward's method minimize in hierarchical clustering?**
    * A) Inertia
    * B) Euclidean Distance
    * C) Sum of squared differences within clusters
    * D) Silhouette Score
    * E) Centroid distance
    * F) Variance Ratio Criterion
47. **Which technique is used to visualize high-dimensional data in 2D?**
    * A) PCA (Principal Component Analysis)
    * B) K-Means++
    * C) Spectral Clustering
    * D) DBSCAN
    * E) Gaussian Mixture Models
    * F) Agglomerative Clustering
48. **Which approach in clustering uses eigenvectors for clustering?**
    * A) K-Means
    * B) DBSCAN
    * C) Spectral Clustering
    * D) Hierarchical Clustering
    * E) Gaussian Mixture Models
    * F) Divisive Clustering
49. **Which algorithm assigns data points to clusters based on their probabilities of belonging to each cluster?**
    * A) DBSCAN
    * B) K-Means
    * C) Spectral Clustering
    * D) Gaussian Mixture Models
    * E) Agglomerative Clustering
    * F) Divisive Clustering
50. **Which metric is commonly used to evaluate clustering performance on ground-truth labeled data?**
    * A) Silhouette Score
    * B) Rand Index
    * C) Inertia
    * D) Dunn Index
    * E) Mahalanobis Distance
    * F) Euclidean Distance

**Answer Key:**

1. C
2. B
3. B
4. D
5. A
6. D
7. B
8. A
9. B
10. B
11. D
12. C
13. C
14. B
15. D
16. C
17. C
18. E
19. A
20. B
21. B
22. A
23. B
24. C
25. C
26. A
27. B
28. D
29. A
30. B
31. B
32. A
33. D
34. B
35. B
36. B
37. B
38. A
39. C
40. C
41. B
42. D
43. D
44. D
45. D
46. C
47. A
48. C
49. D
50. B