Course: MSc DS

Optimisation

Module: 1

Preface

In a world increasingly data-driven, the role of optimisation has grown exponentially in its significance, finding vital applications across various industries including logistics, finance, healthcare, and technology. The "Optimisation" course, a pivotal part of the Master of Science in Data Science curriculum, has been meticulously crafted to provide you with a robust foundation and nuanced understanding of optimisation theories and practices.

This course promises to not only familiarise you with the fundamental concepts like objective functions and decision variables but also venture deep into real-world applications that solve complex problems such as warehouse logistics, assignment predicaments, and knapsack issues. Through an adept mixture of theoretical knowledge and hands-on expertise, the program aims to equip you with the skills needed to navigate and triumph in the diverse landscapes of optimisation problems.

The utilisation of powerful tools like Excel and Python in this course will bridge the gap between theoretical understandings and tangible, real-world applications. As we delve into the practical aspects of the simplex method and its variants, and the creation and interpretation of network analyses, you will be empowered to not only conceptualise solutions but also implement them effectively. This curriculum has been crafted with the perfect blend of expert insights and industry relevance, setting you on the path to becoming a proficient data scientist with a specialisation in optimisation techniques.

Learning Objectives:

1. Grasp the importance and applications of optimisation in data science.
2. Identify different objective functions and their uses in data scenarios.
3. Recognise the role and characteristics of decision variables in optimisation.
4. Gain practical experience with optimisation in Excel and

Python.

5. Compare strengths and weaknesses of both platforms for optimisation.
6. Link decision variables, constraints, and objective functions in optimisation formulation.

Structure:

1.1 Overview of Optimisation in Data Science

Optimisation plays a pivotal role in data science, particularly as it pertains to building, training, and refining models to make accurate predictions or extract meaningful insights from data. At its core, optimisation revolves around finding the best solution from a set of possible solutions.

1.1.1 Historical Evolution of Optimisation Techniques

- **Gradient Descent and Its Variants**: Historically, optimisation started with simple methods like gradient descent, where parameters are updated incrementally in the direction of the steepest decrease in the cost function.
- **Convex Optimisation**: As problems grew more complex, techniques from convex optimisation were adopted to ensure global optimality.
- **Stochastic Methods**: Recognising the computational burden of processing all data at once, stochastic and mini-batch variations of gradient descent became popular, offering a trade-off between computational efficiency and convergence

guarantees.

- **Advanced Optimisers**: With the advent of deep learning, advanced optimisation algorithms such as Adam, RMSprop, and AdaGrad were introduced, providing faster convergence in training deep neural networks.

1.1.2 Real-world Applications of Optimisation in Data Science

- **Recommendation Systems**: Companies like Netflix and Amazon optimise their recommendation algorithms to suggest products or movies that users are most likely to enjoy.
- **Natural Language Processing**: Optimisation helps in refining models like transformers, enabling efficient machine translation, sentiment analysis, and more.
- **Image and Video Analysis**: Optimised deep learning models play a crucial role in tasks such as object detection, facial recognition, and scene segmentation.
- **Operational Research**: Optimisation is applied in logistics for tasks like routing and scheduling to minimise costs and maximise efficiency.

1.1.3 Why is Optimization Crucial for Data Science?

- **Maximising Efficiency and Minimising Loss**: Optimisation ensures that resources (e.g., computational power and memory) are used efficiently, and that the models produce results with the least possible error.
- **Ensuring Robust and Reliable Predictive Models**: A well-optimised model is more likely to generalise well to new, unseen data, making its predictions more reliable.

1.1.4 Challenges and Limitations in Data Science Optimization

- **Overfitting and the Need for Regularization**: One of the most significant challenges in optimisation is overfitting, where a model performs exceptionally well on training data but poorly on new data.

  Regularisation techniques, such as L1 and L2 regularisation, have been introduced to penalise overly complex models, thereby aiding in preventing overfitting.
- **Scalability and High-Dimensional Data**: As data sets grow in size and dimensionality, optimising models becomes

computationally challenging.

Techniques like dimensionality reduction and distributed computing become crucial to manage and optimise models on high-dimensional data.

1.2 Definition and Examples of Objective Functions

An objective function, often denoted as f(x), is a mathematical expression that represents a goal to be achieved in optimisation problems. It quantifies the "goodness" or "quality" of a particular solution in relation to other possible solutions. By evaluating the objective function for various solutions, one can determine which solution is the best, or at least which solutions are better or worse than others.

Characteristics of Objective Functions:

- **Scalar Value:** The objective function should return a scalar value. Regardless of the dimensionality of the input, the output is typically a single number representing the quality of that input.
- **Continuity:** In many optimisation problems, it's desirable (but

not always necessary) for the objective function to be continuous. A continuous function allows for smoother searches in optimisation.

- **Differentiability:** If the function is differentiable, optimisation algorithms that rely on derivatives (like gradient descent) can be used. However, there are optimisation algorithms that work with non-differentiable functions as well.
- **Boundedness:** Some objective functions have known bounds. Knowing the bounds can sometimes simplify the optimisation process or provide insights into the problem.

1.2.1 How Objective Functions Drive the Optimisation Process:

The optimisation process revolves around finding the input (or inputs) that maximise or minimise the objective function. The nature of the objective function (whether it's to be maximised or minimised) dictates the direction of the search. If the goal is to minimise the objective function (like in cost minimisation), the optimiser will seek inputs that lead to the smallest possible value of the function. Conversely, if the goal is to maximise (like in profit maximisation), the optimiser will seek inputs that lead to the largest possible value.

Common Types of Objective Functions in Data Science:

- **Loss Functions (Mean Squared Error, Cross-Entropy, etc.):**
  - Mean Squared Error (MSE): Commonly used in regression problems, it measures the average of the squares of the errors or deviations between predicted and actual values.
  - Cross-Entropy: Used in classification problems, it quantifies the difference between two probability distributions – typically the true label distribution and the predicted distribution.
- **Fitness Functions in Evolutionary Algorithms:**
  - These are functions used to evaluate how close a given solution is to the optimum of the problem. They're crucial in genetic algorithms and other evolutionary computation techniques where potential solutions (often called individuals or organisms) are evaluated

based on their fitness.

Examples of Objective Functions in Action:

- **Minimising Distance in Clustering:**
  - In clustering, the objective is often to minimise the distance between data points in the same cluster while maximising the distance between data points in different clusters. One popular metric is the Sum of Squared Distances (SSD) within clusters.
- **Maximising Likelihood in Statistical Models:**
  - In statistical modelling, the objective can often be to find the parameters that maximise the likelihood of the observed data. This Maximum Likelihood Estimation (MLE) aims to find the parameter values that make the observed data most probable.

1.3 Understanding Decision Variables and their Importance

Decision variables represent the unknowns or the quantities to be determined in an optimisation problem. In mathematical terms, they represent the values you are trying to find to either maximise or minimise an objective function.

For example, in a linear programming problem where you're trying to maximise profit by determining the number of units of two products to produce, the number of units of each product will be your decision variables.

1.3.1 Characteristics and Types of Decision Variables: Decision variables can have various characteristics, such as:

- Continuous or Discrete: Continuous variables can take on any value within a defined range (e.g., the amount of resources allocated), while discrete variables can only take on specific values, often integers (e.g., the number of machines).
- Binary or Integer: Binary variables have only two possible values, often 0 or 1, indicating the absence or presence of a particular feature. Integer variables can take whole numbers within a range.

1.3.2 The Role of Decision Variables in Formulating Optimisation Problems: Decision variables play a pivotal role in:

- **Objective Function Formulation:** Your objective, whether it's to maximise or minimise, is expressed as a function of your decision variables.

- **Representing Real-world Decisions:** Decision variables embody the choices you make in real-world scenarios, e.g., the amount of investment in various sectors, or the quantity of a product to manufacture.

1.3.3 Importance of Decision Variables in Data Science:

- **Influencing the Behaviour of Predictive Models:** Decision variables, when used in predictive modelling, can significantly influence the behaviour of the model. The values they take can guide the model to make specific predictions.
- **Determining the Feasibility and Solution Space of Optimisation Problems:** The range and types of decision variables outline the feasible region of solutions. This space can either be finite or infinite, and the nature of decision variables often determines the methods used for optimisation.

1.3.4 Interplay between Decision Variables and Constraints: Constraints in optimisation problems define the feasible region within which solutions can exist. The relationship between decision variables and constraints is integral.

- Feasibility: A solution is considered feasible if it satisfies all constraints.
- Influencing Decisions: Constraints can limit or expand the range of values that decision variables can take.

1.3.5 Bounding and Limiting Decision Variable Values: Bounding involves setting upper and/or lower limits on the values that decision variables can assume.

- Upper Bounds: The maximum allowable value for a decision variable.
- Lower Bounds: The minimum allowable value for a decision variable.

This is crucial to guide the optimisation problem toward realistic solutions and to prevent decision variables from taking on extreme or non-meaning values.

1.3.6 How Constraints Shape Optimisation Outcomes: The final solution of an optimisation problem is not just influenced by the objective function, but also significantly by constraints.

- **Defining Solution Space:** Constraints define the solution space by creating boundaries within which decision variables can vary.
- **Affecting Optimal Solutions:** Constraints can either open up new areas of optimality or restrict certain solutions from being optimal.
- **Introducing Trade-offs:** When multiple constraints interact, there may be necessary trade-offs to be made among decision variables to ensure all constraints are met.

1.4 Hands-on: Defining Objective Functions and Decision Variables in Excel and Python

1. **Setting up an Optimisation Problem in Excel**

- **Inputting Data and Defining Variables**: Excel is a widely used tool for setting up and solving basic optimisation problems. The first step is inputting your data into cells, ensuring it's organised and readily accessible.

Variables, or decision variables, represent the aspects of the problem we have control over and are trying to optimise. In Excel, these are typically defined in dedicated cells to track their values.

- **Using Excel's Solver Tool for Optimization**: Solver is an Excel add-in that allows for the definition and solving of optimisation problems. Once your data and decision variables are set, you can specify the objective function (i.e., what you are trying to minimise or maximise) and constraints (i.e., the boundaries or conditions that the solution must satisfy).

  The Solver will then use various algorithms to find the optimal solution within the constraints.

2. **Optimisation with Python**

- **Introduction to Libraries: SciPy, CVXPY, and more**: Python, being a powerful programming language, has numerous libraries for optimisation. Among the most popular are:

  - *SciPy:* A library for mathematics, science, and engineering, which includes functions for optimisation.

- *CVXPY:* A library specifically designed for convex optimisation problems.

  The choice of library often depends on the specific type of optimisation problem, as some libraries may be more suitable or efficient for certain problem structures.

- **Defining and Solving Optimisation Problems Using Python**
  - o Once the appropriate library is chosen, the process begins by defining decision variables in the programming environment.
  - o The next step is to define the objective function in terms of these variables. Constraints are also encoded similarly.
  - o Python libraries usually provide functions to 'solve' or 'optimise' the defined problem, producing an optimal solution and related metrics.

3. **Comparative Analysis: Excel vs. Python in Optimisation**

- **Strengths and Weaknesses of Each Platform**

Excel:

- **Strengths:** User-friendly interface, no need for advanced programming skills, suitable for small to medium-sized problems, and visualisation capabilities.
- **Weaknesses:** Not ideal for very large datasets, limited in terms of advanced optimisation algorithms, and may be computationally slower for complex problems.

Python:

- **Strengths:** Highly scalable, wide range of libraries and algorithms, suitable for simple to very complex optimisation tasks, and robust for large datasets.
- **Weaknesses:** Requires programming knowledge, may have a steeper learning curve, and visualisation can be less straightforward than in Excel.

- **Recommendations for Different Optimisation Scenarios**

- For small to medium datasets and simple linear problems, Excel may suffice due to its ease of use and quick setup.
- For large datasets, non-linear problems, or tasks that require repeated or automated optimisation, Python is the preferable platform. It offers a more scalable, flexible, and powerful environment for handling complex optimisation scenarios.

1.5 Summary

- Mathematical expressions that quantify the performance or suitability of solutions in an optimisation problem, such as minimising errors or maximising likelihoods.
- **Decision Variables** are the controllable parameters on which the outcomes of the optimisation problem depend. They determine potential solutions and directly influence the behaviour of data-driven models.
- Within the data science realm, optimisation ensures models are robust, avoids issues like overfitting, and tailors solutions

to specific problems, leading to enhanced accuracy and efficiency.

❖ Platforms like Excel provide built-in tools for optimisation, while programming languages like Python offer extensive libraries such as SciPy and CVXPY, catering to complex optimisation problems.

❖ The real-world limitations or restrictions that bound the possible solutions. These constraints, combined with decision variables, shape the feasible region of optimisation problems.

1.6 Keywords

- **Optimisation**: At its core, optimisation is the process of finding the best solution or outcome from a set of possible choices. In data science, optimisation often refers to techniques employed to adjust models and algorithms to improve their performance or accuracy based on a specific criterion, such as reducing error or maximising efficiency.
- **Objective Function**: An objective function (often also called a loss function, cost function, or fitness function) quantifies

how well a solution meets the desired objective. In data science, the objective function often evaluates the performance of a model or algorithm. For instance, a common objective function is the Mean Squared Error which measures the average squared differences between predicted and actual values.

- **Decision Variables**: Decision variables represent the unknowns or the parameters to be determined in an optimisation problem. In the context of a data science problem, decision variables could represent weights in a neural network, coefficients in a regression model, or any tunable parameter that can influence the outcome.
- **Overfitting**: Overfitting occurs when a model learns the training data too closely, including its noise and outliers, which makes it perform poorly on new, unseen data. It essentially captures patterns that are not generalisable. Regularisation techniques and optimisation play a crucial role in mitigating overfitting.

- **Regularisation**: Regularisation introduces a penalty on the complexity of a model, helping to prevent overfitting. It works by adding a term to the objective function that penalises certain model parameters if the model becomes too complex. Common regularisation techniques in data science include L1 (Lasso) and L2 (Ridge) regularisation.
- **Solver Tool (in Excel)**: The Solver Tool in Excel is an optimisation add-in that allows users to specify certain criteria and constraints and then find the best solution to an optimisation problem. This might involve maximising a particular value, minimising costs, or finding a target value in a specific way.

1.7 Self-Assessment Questions

1. How does optimisation play a crucial role in ensuring the efficiency and reliability of predictive models in data science?
2. What are the key characteristics that define an objective function, and why are they essential in the optimisation process?

3. Which type of objective function, among Mean Squared Error and Cross-Entropy, would be more suitable for classification problems in data science and why?
4. What is the significance of decision variables in determining the feasibility and solution space of optimisation problems?
5. Which platform, between Excel and Python, offers more flexibility and scalability when handling complex optimisation problems, and why?

1.8 Case Study

Title: Optimisation of Inventory Management for a Japanese Sake Brewery

Introduction:

In Kyoto, there’s a century-old sake brewery, Nishiyama Shuzo. As demand for its artisanal sake grew, both domestically and internationally, the brewery faced an inventory management challenge. Given the intricate process of brewing sake, which involves multiple ingredients and stages, maintaining an optimised inventory was crucial to ensure a continuous and efficient production process without wasting resources.

Background:

Nishiyama Shuzo's primary challenge was optimising the quantities of rice, water, and yeast for different sake variants, considering the seasonal demand variations and the fermentation times of each type. Additionally, space within their traditional brewing facilities was limited, making it critical to avoid overstocking.

The brewery employed a data science team that used predictive analytics to forecast demand patterns, including seasonal peaks during festivals and tourism influxes. They combined this with a constrained optimisation algorithm to determine the best quantities of ingredients to keep on hand at different times of the year. By analysing past sales data, tourism patterns, and even weather forecasts, the algorithm could suggest optimal purchasing strategies for the brewery.

The results were impressive. Nishiyama Shuzo reduced its inventory costs by 20% in the first year and minimised wastage of ingredients, especially premium-grade rice with a limited shelf life. The optimised inventory system also ensured that they never ran out of stock during peak demand, enhancing customer satisfaction.

Questions:

1. How did seasonal demand variations affect the inventory management at Nishiyama Shuzo?
2. Which data sources did the data science team incorporate to optimise the purchasing strategies for the brewery?
3. What were the significant outcomes achieved by Nishiyama Shuzo after implementing the optimised inventory system?

1.9 References

1. "Convex Optimisation" by Stephen Boyd and Lieven Vandenberghe
2. "Numerical Optimisation" by Jorge Nocedal and Stephen J. Wright
3. "Introduction to Linear Optimization" by Dimitris Bertsimas and John N. Tsitsiklis
4. "Optimisation for Machine Learning" edited by Suvrit Sra, Sebastian Nowozin, and Stephen J. Wright

5. "The Nature of Computation" by Cristopher Moore and Stephan Mertens