Course: MSc DS

Optimisation

Module: 3

Learning Objectives:

1. Grasp the role and impact of constraints in optimisation.
2. Differentiate between hard and soft constraints.
3. Apply constraints to practical optimisation scenarios.
4. Implement constraints using Excel and Python.
5. Assess how constraints affect optimisation results.
6. Recognise pitfalls and best practices in constraint formulation.

Structure:

3.1 Understanding the Role of Constraints in Optimisation Problems

Optimisation refers to the art and science of choosing the best solution from a set of feasible solutions. Two of the most critical components of any optimisation problem are the objective function and the constraints.

- **Objective Function**: This is the function that needs to be optimised. It can either be maximised (like profit) or minimised (like cost). The objective function quantifies the goal of the optimisation problem and provides a metric against which the quality of a solution is judged.
- **Constraints**: These define the feasible region within which a solution must lie. Constraints can take various forms – inequalities, equalities, or even more complex relations. They serve to restrict and guide the set of possible solutions to a manageable and practical subset.

3.1.1 Importance of Constraints in Shaping Solutions

The significance of constraints in an optimisation problem cannot be overstated. While the objective function delineates what we aspire to achieve, it is the constraints that give structure to our problem.

- Shape of the Solution Space: Constraints often define the geometry of the feasible solution space. For example, linear constraints give rise to polyhedral solution spaces in linear programming.
- Influence on Optimal Solutions: Constraints can lead to multiple optimal solutions or even the absence of any feasible solutions. In some cases, tightening or loosening a constraint can shift the optimal solution significantly.

3.1.2 The Interplay Between Constraints and Feasibility

The delicate balance between constraints and feasibility is central to the concept of optimisation.

- Feasible Region: All constraints collectively define the feasible region, which is the set of all solutions that satisfy all the constraints.
- Infeasibility: If there exists no solution that can satisfy all

constraints simultaneously, the problem is termed infeasible. Recognising infeasibility early can save computational resources and effort.

- Tightness of Constraints: Overly restrictive constraints can make a problem infeasible or can lead to a solution that is suboptimal in practice. On the other hand, very loose constraints might yield solutions that, while optimal on paper, are not practically useful.

3.1.3 Real-world Implications of Ignoring Constraints

Neglecting constraints or not treating them with due seriousness can lead to various real-world ramifications.

- Impractical Solutions: Ignoring constraints might produce a solution that is theoretically optimal but is not implementable in a real-world scenario. For example, recommending production levels that exceed machinery capacity.
- Economic Implications: In business contexts, overlooking constraints can result in substantial economic losses. For instance, underestimating budget constraints in project

management can lead to overspending.

- Safety Concerns: In sectors like engineering and medicine, ignoring constraints can lead to safety issues. Consider designing a bridge without considering weight constraints.

3.2 Types of Constraints: Hard vs Soft

Hard constraints represent non-negotiable boundaries or conditions that a solution must satisfy. Violating a hard constraint renders a solution infeasible, regardless of the potential value or benefits of the solution. In other words, hard constraints define what is permissible in the solution space.

Example: In scheduling optimisation for an airline, ensuring that the minimum rest time between flights for pilots is maintained can be considered a hard constraint. This is because violating it would not only breach labour regulations but also compromise safety.

3.2.1 Soft Constraints: A Matter of Preference and Flexibility

Soft constraints, on the other hand, indicate preferences rather than strict requirements. While solutions can violate these constraints, they typically incur a penalty or are considered less optimal than solutions that adhere to them.

Example: In the same airline scheduling problem, minimising the time aircraft spend on the ground (turnaround time) can be considered a soft constraint. While it's preferable to have shorter turnaround times for efficiency, exceeding a target turnaround time doesn't render a schedule infeasible.

3.2.2 Comparative Analysis: When to Use Hard vs. Soft Constraints

- Nature of the Problem: Hard constraints are typically derived from regulations, physical limitations, or conditions that ensure the viability of a solution. Soft constraints are often tied to objectives that enhance the quality or desirability of a solution.
- Flexibility vs. Strictness: Hard constraints offer no room for negotiation, whereas soft constraints allow decision-makers to evaluate trade-offs. If there is a measure of flexibility or if decision-makers are willing to accept certain penalties, soft constraints can be utilised.
- Penalty Consideration: Soft constraints usually involve a

penalty for violation, which can be incorporated into the objective function in optimisation models. Hard constraints don't have this flexibility.

3.2.3 Benefits and Limitations of Each Constraint Type

Hard Constraints:

- **Benefits**:
    - Ensures essential conditions are met.
    - Provides clarity on what constitutes an infeasible solution.
- **Limitations**:
    - Can significantly reduce the feasible solution space.
    - Might lead to no feasible solutions in certain scenarios.

Soft Constraints:

- **Benefits**:
    - Offers flexibility and allows exploration of a wider solution space.
    - Provides a mechanism to balance multiple competing objectives.

- **Limitations**:
    - Can introduce ambiguity, especially when multiple soft constraints conflict.
    - Requires careful calibration of penalty terms to ensure they reflect real-world trade-offs.

3.2.4 From Theory to Practice: Merging Both Constraint Types

In real-world optimisation problems, a mix of both hard and soft constraints is often encountered. Merging both constraint types requires:

- **Model Formulation**: Begin by identifying and incorporating all hard constraints to ensure all solutions are within the permissible space.
- **Penalty Integration**: Integrate soft constraints, typically by adding penalty terms to the objective function. This allows the model to rank solutions based on the degree to which they satisfy soft constraints.
- **Trade-off Analysis**: During post-optimisation analysis, evaluate trade-offs made with respect to soft constraints. It's

crucial to ensure that the penalties reflect actual business or real-world values.

- **Iterative Refinement**: Based on insights from solutions and the observed trade-offs, adjust the optimisation model, especially the penalties associated with soft constraints, and solve again. This iterative process helps in refining solutions further.

3.3 Applying Constraints to Real-World Problems: Examples and Case Studies

3.3.1 Case Study 1: Transportation Optimisation – Balancing Cost and Emission Constraints

Transportation optimisation often involves determining the most efficient means to move goods from one point to another. Within the realm of sustainability and corporate responsibility, companies are faced with the challenge of balancing economic costs and environmental impacts.

- **Objective**: Minimise total transportation costs, while also ensuring that CO2 emissions are below a specific threshold.

- **Variables**:
  - Modes of transportation (e.g., truck, train, air).
  - Routes selected.
  - Frequency of shipments.
- **Constraints**:
  - Maximum allowable CO_2 emissions.
  - Delivery deadlines.
  - Transportation capacity (e.g., tonnage that can be transported via a specific mode).

3.3.2 Case Study 2: Workforce Scheduling – Managing Employee Preferences and Shift Requirements

Employee scheduling is a common operational challenge for businesses. The optimisation problem revolves around assigning shifts to employees in a manner that meets operational requirements and respects employee preferences to the greatest extent possible.

- **Objective**: Assign shifts to employees to maximise operational efficiency and employee satisfaction.

- **Variables**:
  - o Employee availability.
  - o Employee skill set and role.
  - o Shift requirements (e.g., length, time of day).
- **Constraints**:
  - o Minimum and maximum hours per employee.
  - o Rest periods between shifts.
  - o Specific skill set requirements for particular shifts.

3.3.3 Practical Example: Investment Portfolio Selection with Budgetary and Risk Constraints

Investing is fundamentally about maximising returns while managing risk. An optimised portfolio is a balance between the assets selected, the proportion of each asset in the portfolio, and the overall risk tolerance of the investor.

- **Objective**: Maximise expected portfolio return, given a certain risk tolerance.
- **Variables**:
  - o Selection of assets (e.g., stocks, bonds, real estate).

- Allocation percentages of each asset.

- **Constraints**:
    - Budget constraints (e.g., total investment amount).
    - Risk thresholds (often quantified using measures like Value at Risk or Standard Deviation).
    - Sector allocation limits (e.g., no more than 30% in tech stocks).

3.3.4 Real-world Considerations: Dealing with Dynamic Constraints

In real-world scenarios, constraints are rarely static. Prices fluctuate, environmental regulations change, and unforeseen events can introduce new variables or constraints into the optimisation problem.

- **Adaptive Optimisation**: Modern optimisation techniques must be adaptive, capable of adjusting to new information and recalibrating solutions as constraints change.
- **Scenario Analysis**: By simulating different scenarios, decision-makers can understand how solutions might change

under different future conditions. For example, how would a transportation plan change if fuel prices doubled, or if a new regulation capped weekly driving hours?

- **Real-time Monitoring & Feedback**: With the proliferation of IoT devices and real-time data analytics, optimisation solutions can be continuously updated based on live feedback. This enables systems to be more responsive and resilient.

3.4 Summary

❖ Constraints dictate the boundaries within which an optimisation problem must find a solution. They ensure that solutions are feasible and adhere to specific criteria or limitations, shaping the space where optimal solutions reside.

❖ Hard Constraints are non-negotiable boundaries. If a solution violates a hard constraint, it is deemed infeasible. Soft Constraints are flexible boundaries, often representing preferences rather than strict rules. Solutions can violate soft constraints, typically at some cost.

- In real-world scenarios, constraints represent practical limitations or conditions that solutions must adhere to. Examples include budget limits in finance, machine capacities in manufacturing, or delivery timeframes in logistics.
- Excel offers tools like the Solver that allow users to define and solve optimisation problems. Users can specify objective functions and constraints to find optimal solutions within set boundaries.
- Python, with libraries like SciPy, provides robust capabilities for defining and solving optimisation problems. Users can create objective functions, specify hard and soft constraints, and utilise powerful algorithms to find optimal solutions.
- While Excel provides a user-friendly interface suitable for smaller optimisation problems and quick analyses, Python offers more flexibility, scalability, and is better suited for complex, large-scale problems.

3.5 Keywords

- **Optimisation**: Optimisation refers to the process of determining the best solution from a set of feasible solutions. In mathematical terms, it involves either maximising or minimising a certain objective function, subject to given constraints. For example, in business, it can relate to maximising profits while minimising costs.

- **Objective Function**: The objective function is the primary goal of an optimisation problem. It quantifies the outcome we're trying to achieve, whether that's to maximise (like profits) or minimise (like costs). Essentially, it's a mathematical representation of what needs to be optimised.
- **Hard Constraints**: Hard constraints are non-negotiable limits or restrictions in an optimisation problem. When these constraints are violated, the solution is considered infeasible or unacceptable. For instance, a budget limit for a project can be a hard constraint, as going over it might not be allowed.
- **Soft Constraints**: Unlike hard constraints, soft constraints are flexible. While they represent desired conditions, they can be violated if necessary, though often at some cost or penalty. An

example could be a preferred delivery date for a project. While it's ideal to meet that date, delaying by a day or two might still be acceptable but less than ideal.

- **Feasibility**: In the context of optimisation, feasibility refers to the set of solutions that satisfy all the given constraints of the problem. A feasible solution does not violate any hard constraints and is considered an acceptable answer to the optimisation problem, even if it might not be the optimal one.
- **Solver Tool (in Excel)**: Solver is an Excel add-in that provides a user interface for specifying and solving optimisation problems. It allows users to define an objective function, decision variables, and constraints, and then utilises various algorithms to find the best solution. It's especially handy for simpler optimisation problems and those who might not have extensive coding experience.

3.6 Self-Assessment Questions

1. How do constraints influence the feasible solution space in optimisation problems?

2. What distinguishes hard constraints from soft constraints in optimisation scenarios?
3. Which of the following is NOT typically a consideration when implementing constraints in real-world optimisation problems: a) Employee Preferences, b) Risk Assessment, c) Colour Preferences of a CEO for Graph Visualisation, d) Budgetary Limits?
4. What are the primary steps to set up an Excel spreadsheet for optimisation using the Solver Tool?
5. Which platform, between Excel and Python's SciPy, would be more suitable for handling large-scale, complex optimisation problems, and why?

3.7 Case Study

Title: Tokyo Metro's Rush Hour Optimization

Introduction:

The Tokyo Metro is one of the busiest metropolitan transport systems in the world. With over 3.2 billion passenger rides annually, the efficient and timely management of this system is crucial for the daily life and economy of Tokyo. One of the most significant challenges the Tokyo Metro faces is the morning rush hour congestion, with trains operating beyond 200% capacity.

Background:

In 2019, a project initiated by the Tokyo Metropolitan Government aimed to optimise this congestion by manipulating two primary variables: train schedules and passenger incentives. An advanced data analytics team, consisting of top Data Science graduates, was assembled to address this issue.

First, historical data, including train schedules, passenger counts, and boarding times, were analysed. The objective was to identify bottlenecks and potential time slots where adding or rescheduling a train could alleviate pressure. After extensive modelling, the team proposed slight alterations to the schedule, strategically placing them during peak congestion times.

Secondly, the team suggested a 'Shifted Work Hour' incentive. Companies in Tokyo were encouraged to allow flexible working hours. Employees who avoided peak hours were provided with

Metro vouchers and discounted coffee at partnering cafes. This initiative aimed to distribute the passenger load more evenly throughout the morning.

By the end of 2020, the Tokyo Metro saw a 15% reduction in congestion during peak hours, which not only improved the passenger experience but also reduced wear and tear on the infrastructure, leading to long-term economic savings.

Questions:

1. What two primary variables were manipulated to address the rush hour congestion in the Tokyo Metro system?
2. How did the Tokyo Metro incentivise passengers to travel outside of peak hours?
3. What long-term benefits, apart from decongestion, did the Tokyo Metro gain from this optimisation project?

3.8 References

1. "Convex Optimisation" by Stephen Boyd and Lieven Vandenberghe
2. "Introduction to Linear Optimization" by Dimitris Bertsimas

and John N. Tsitsiklis

3. "The Art of Computer Programming, Volume 2: Seminumerical Algorithms" by Donald E. Knuth
4. "Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking" by Foster Provost and Tom Fawcett
5. "Numerical Optimisation" by Jorge Nocedal and Stephen J. Wright