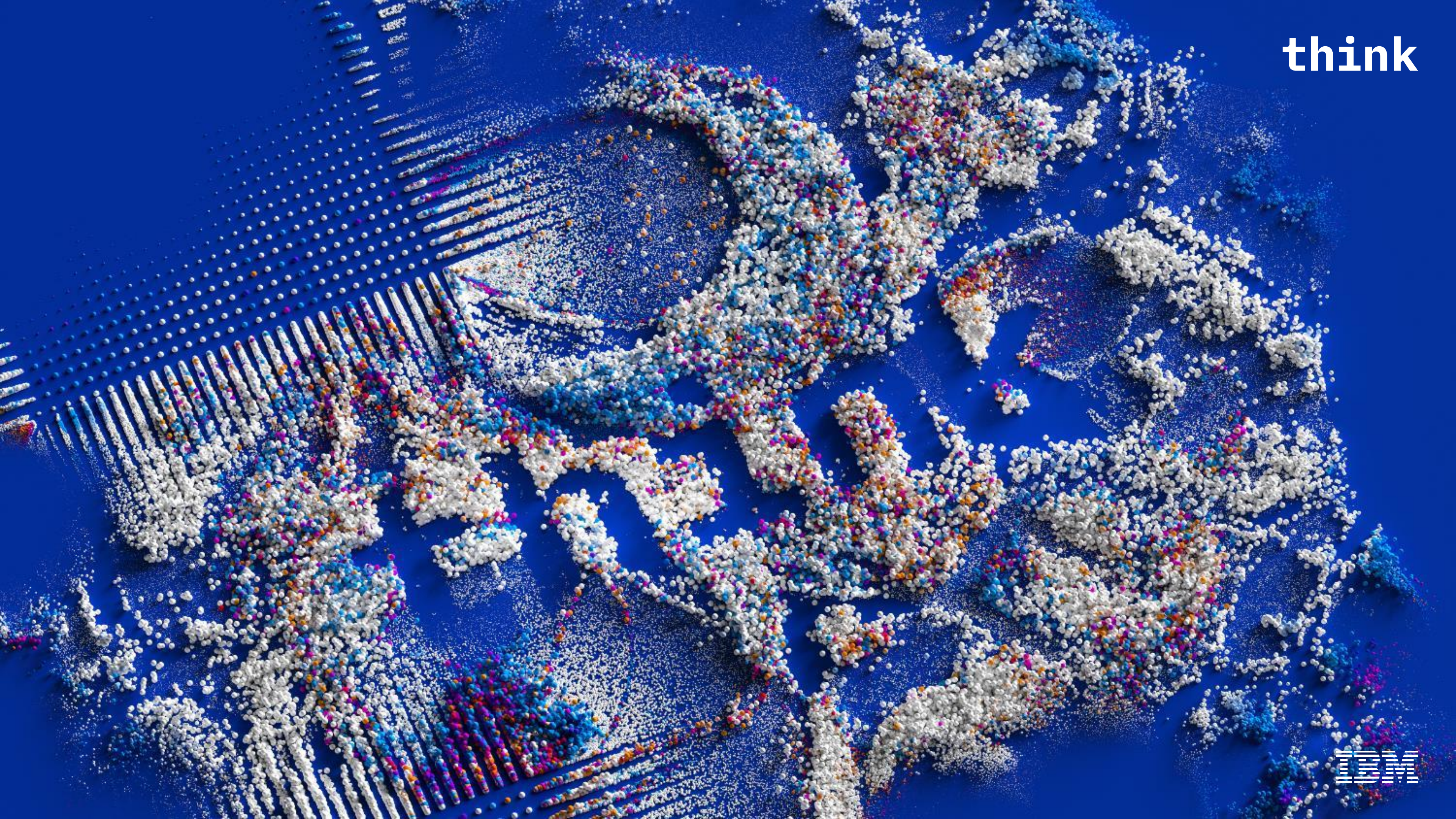

think

Actionable Decisions

from

Data Science

think

IBM

Agenda

Academic Context

Examples

Decision Optimization & Machine Learning

A bit more technology

Hands-on

Optimization

think

An act, process, or methodology of making something (such as a design, system, or decision) as fully perfect, functional, or effective as possible

Specifically:

the mathematical procedures (such as finding the maximum of a function) involved in this

https://www.merriam-webster.com/dictionary/optimization

IBM

Academic context

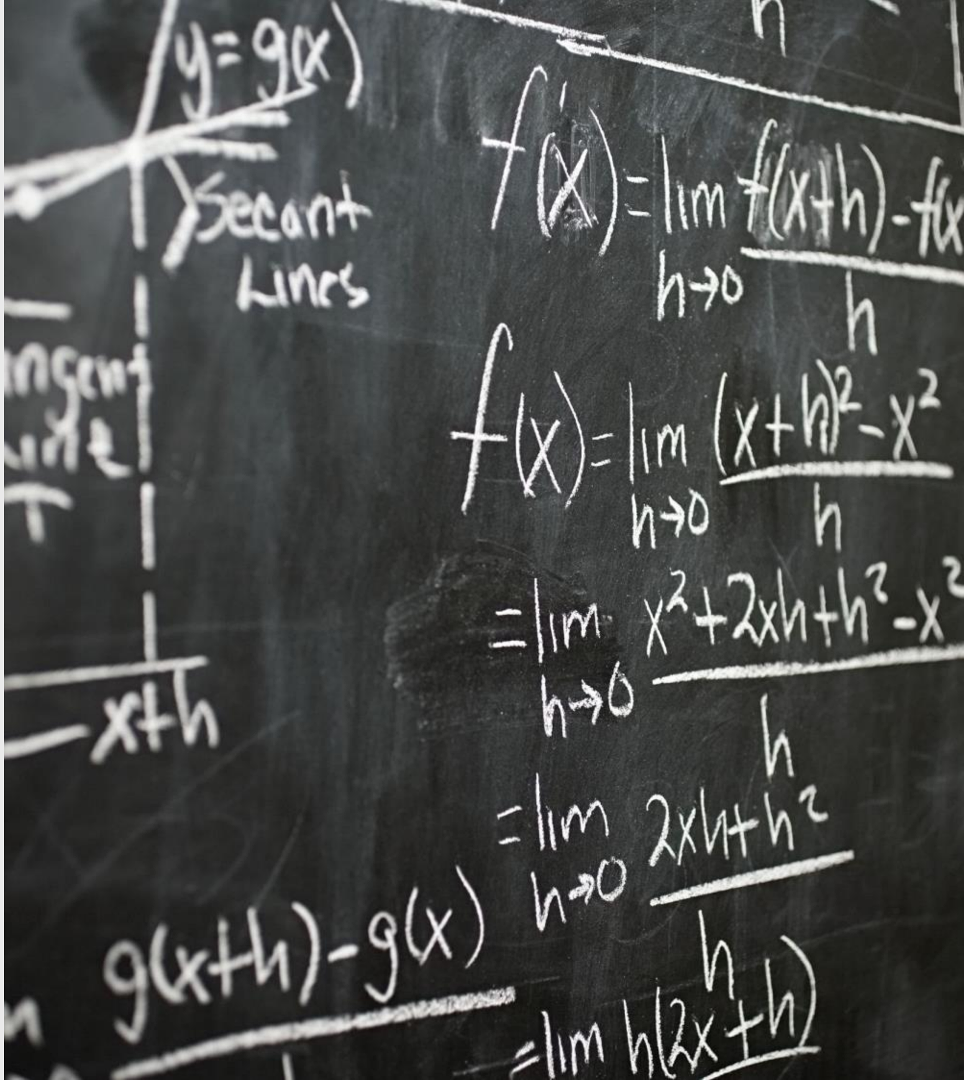
Mathematical Optimization

$$\min f(x)$$

where

$$x \in A \subseteq \mathbb{R}^n$$

$$f : A \to \mathbb{R} \text{ continuous}$$



Academic context

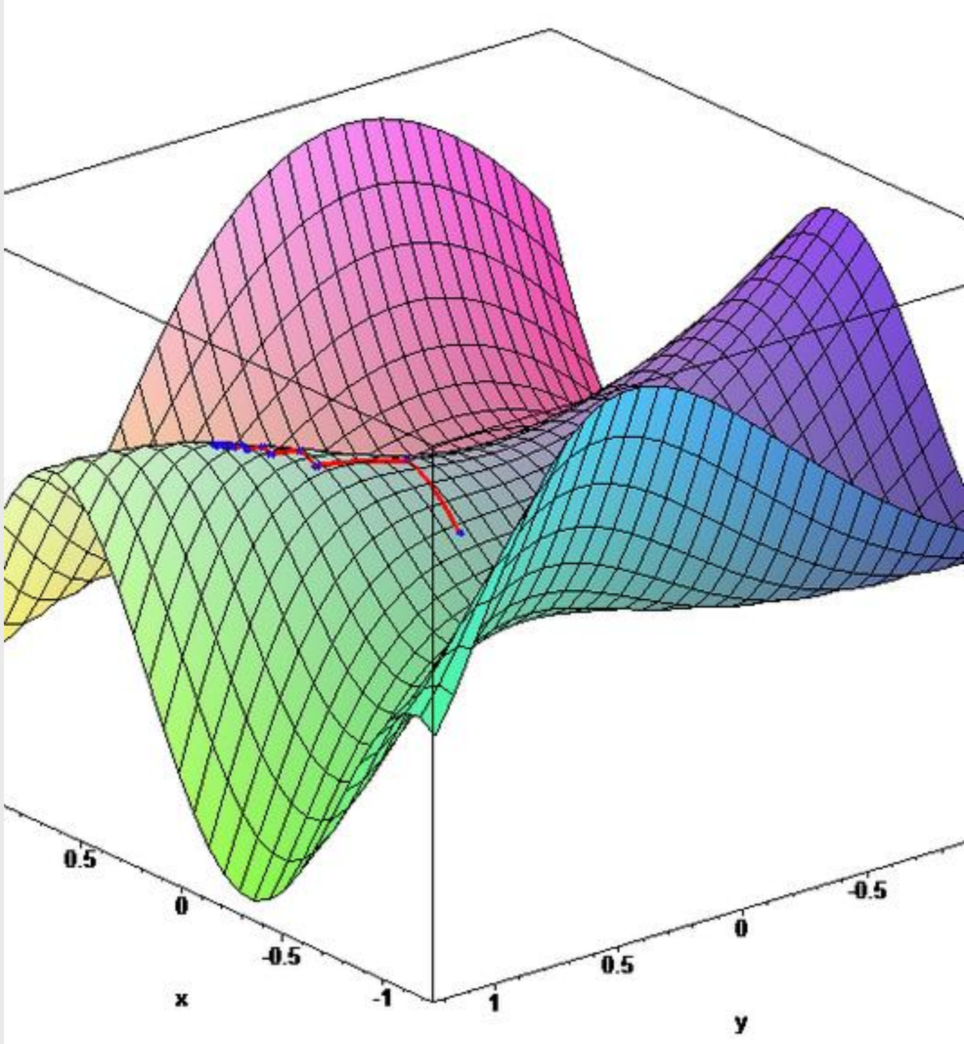
Mathematical Optimization

$$\min f(x)$$

where

$$x \in A \subseteq \mathbb{R}^n$$

$$f : A \to \mathbb{R} \text{ differentiable}$$



Academic context

Mathematical Optimization

$$\min f(x)$$

s.t.

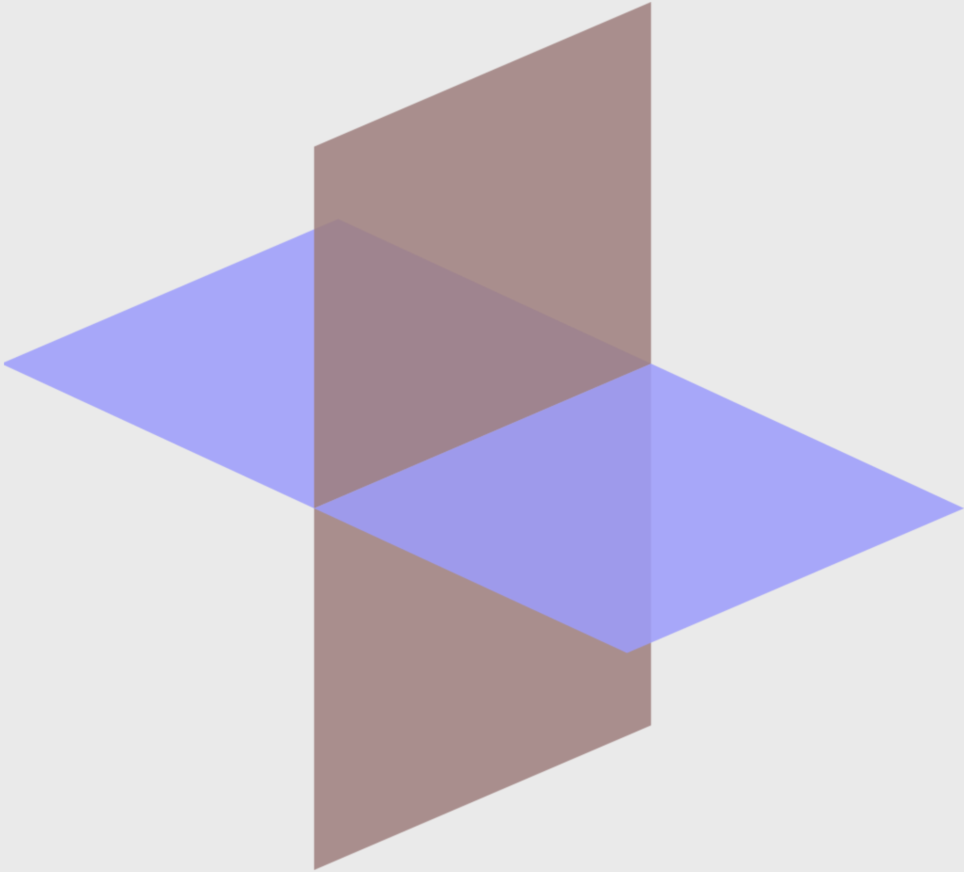
$$g_i(x) \leq 0$$
$$h_j(x) = 0$$

where

$$x \in \mathbb{R}^n$$

$$f : \mathbb{R}^n \to \mathbb{R} \text{ linear}$$

$$g_i : \mathbb{R}^n \to \mathbb{R} \text{ linear}$$
$$h_j : \mathbb{R}^n \to \mathbb{R} \text{ linear}$$



Academic context

Mathematical Optimization

$$\min c^T x$$

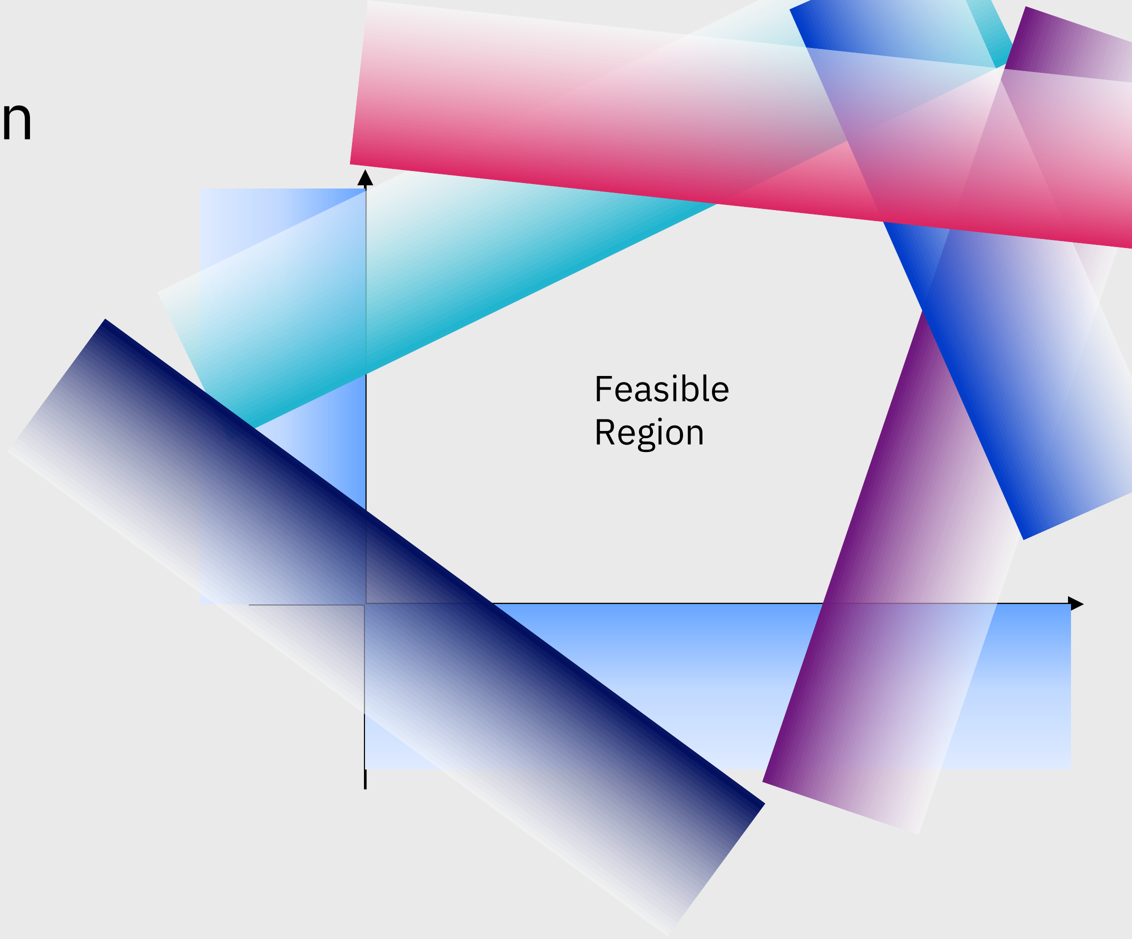
s.t.

$$Ax \geq b$$
$$x \geq 0$$

where

$$x \in \mathbb{R}^n$$
$$A \in \mathbb{R}^{m*n}$$
$$c \in \mathbb{R}^n$$
$$b \in \mathbb{R}^m$$

"Linear Program"



Academic context

Mathematical Optimization

$$\min c^T x$$

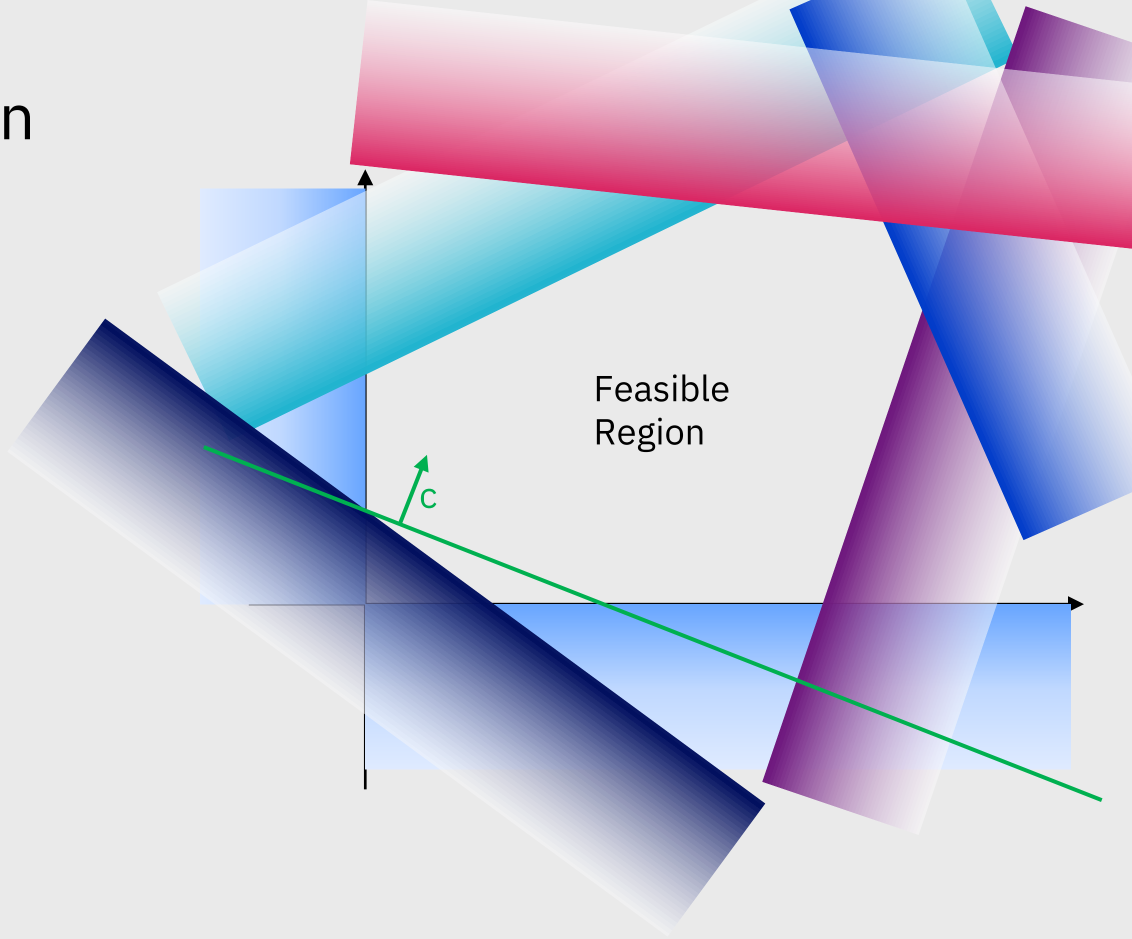
s.t.

$$Ax \geq b$$
$$x \geq 0$$

where

$$x \in \mathbb{R}^n$$
$$A \in \mathbb{R}^{m*n}$$
$$c \in \mathbb{R}^n$$
$$b \in \mathbb{R}^m$$

Linear Program



Academic context

Mathematical Optimization

$$\min \; c^T x$$

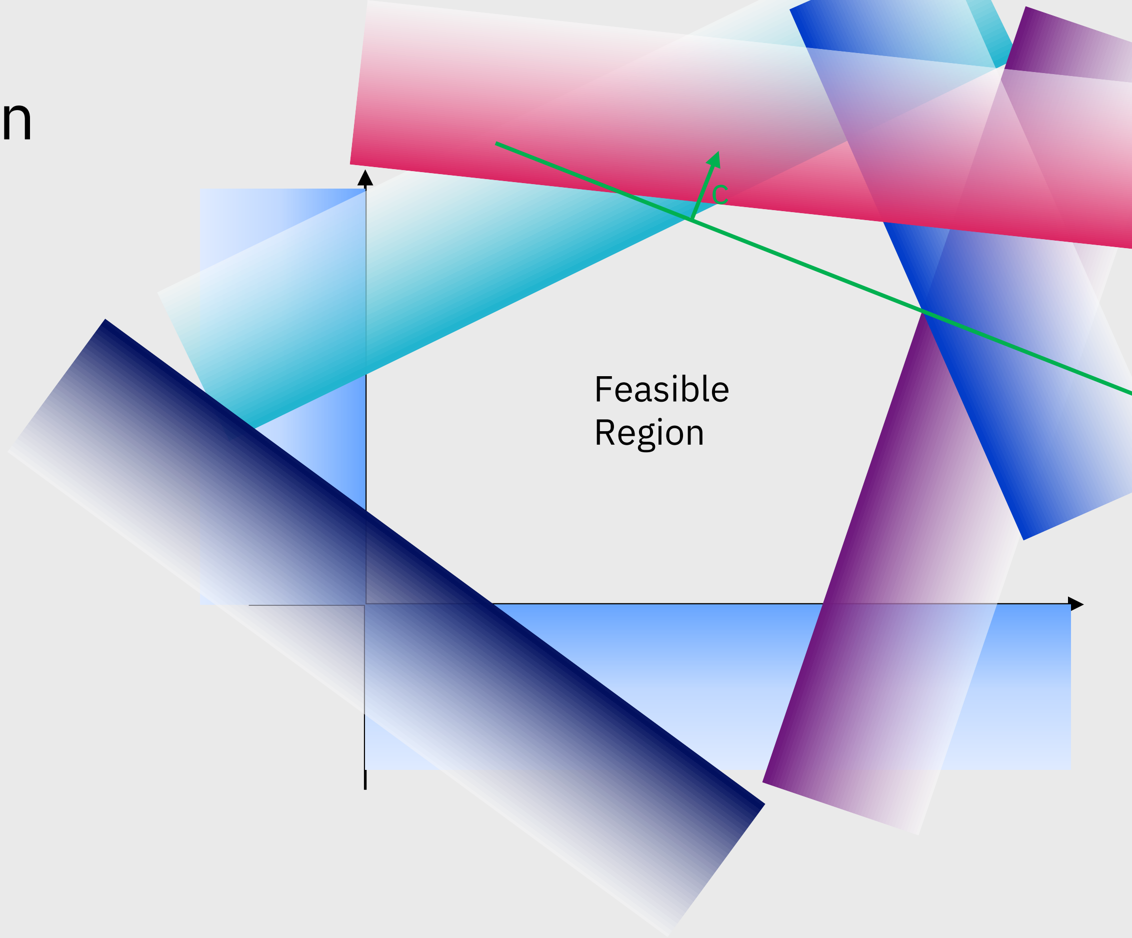
s.t.

$$Ax \geq b$$
$$x \geq 0$$

where

$$x \in \mathbb{R}^n$$
$$A \in \mathbb{R}^{m*n}$$
$$c \in \mathbb{R}^n$$
$$b \in \mathbb{R}^m$$

Linear Program



Academic context

Mathematical Optimization

$$\min c^T x$$

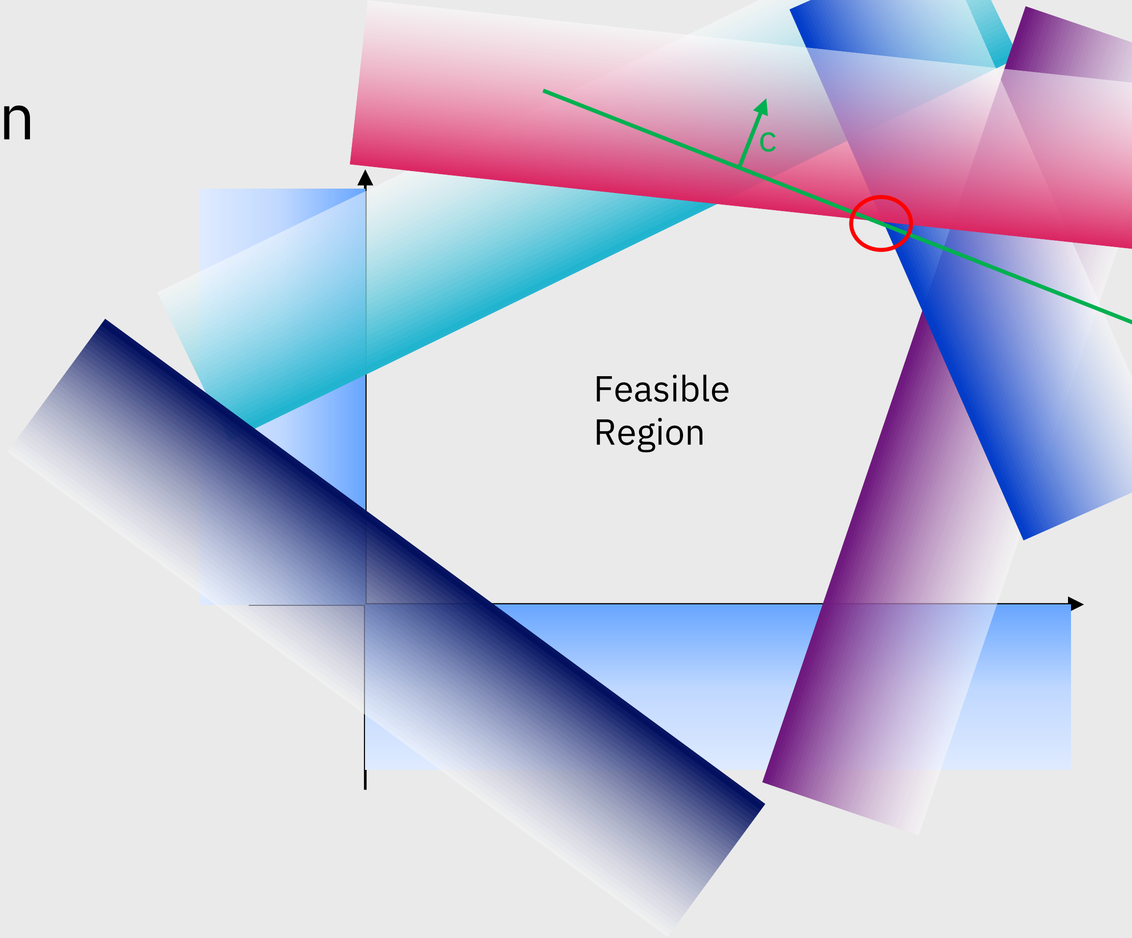
s.t.

$$Ax \geq b$$
$$x \geq 0$$

where

$$x \in \mathbb{R}^n$$
$$A \in \mathbb{R}^{m*n}$$
$$c \in \mathbb{R}^n$$
$$b \in \mathbb{R}^m$$

Linear Program



Academic context

Mathematical Optimization

$$\min c^T x$$

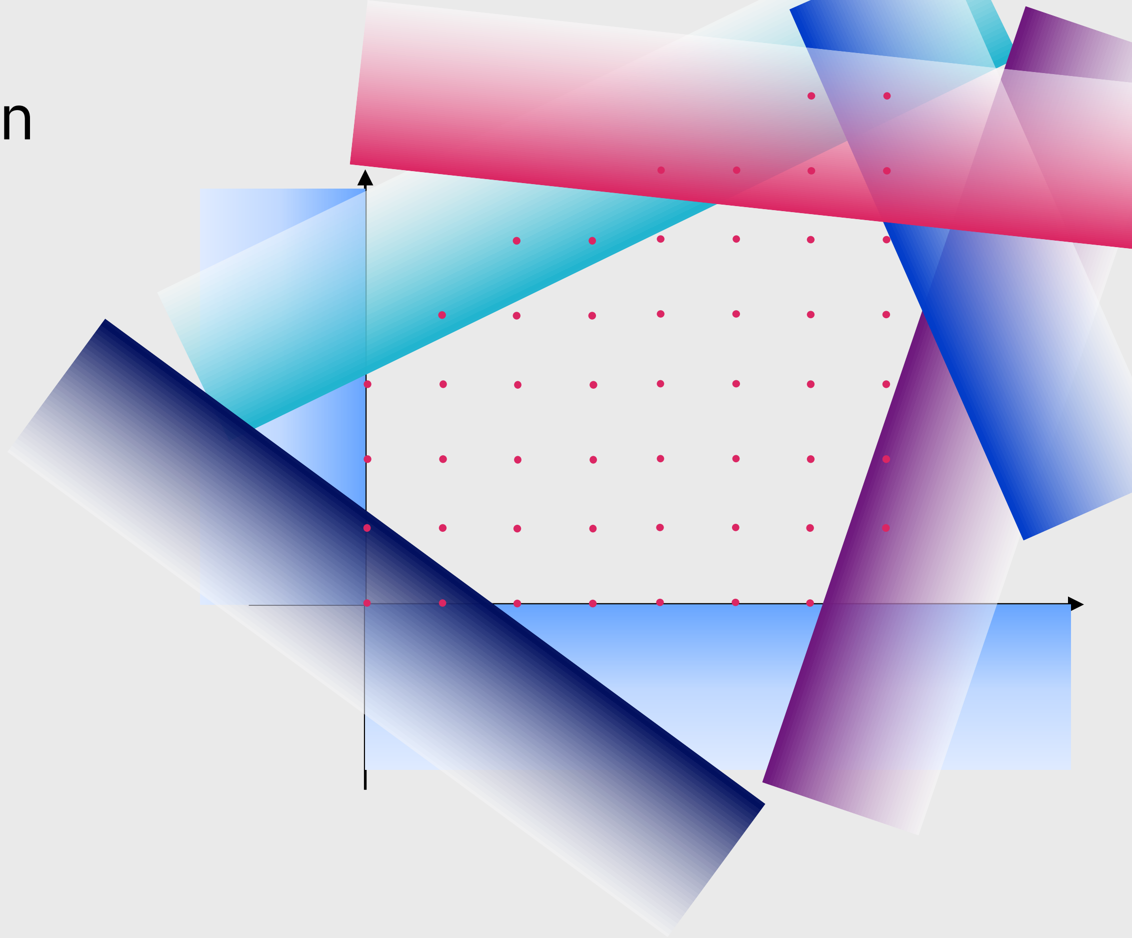
s.t.

$$Ax \geq b$$
$$x \geq 0$$

where

$$x \in \mathbb{Z}^n$$
$$A \in \mathbb{R}^{m*n}$$
$$c \in \mathbb{R}^n$$
$$b \in \mathbb{R}^m$$

Integer Linear Program



Academic context

Mathematical Optimization

$$\min c^T x$$

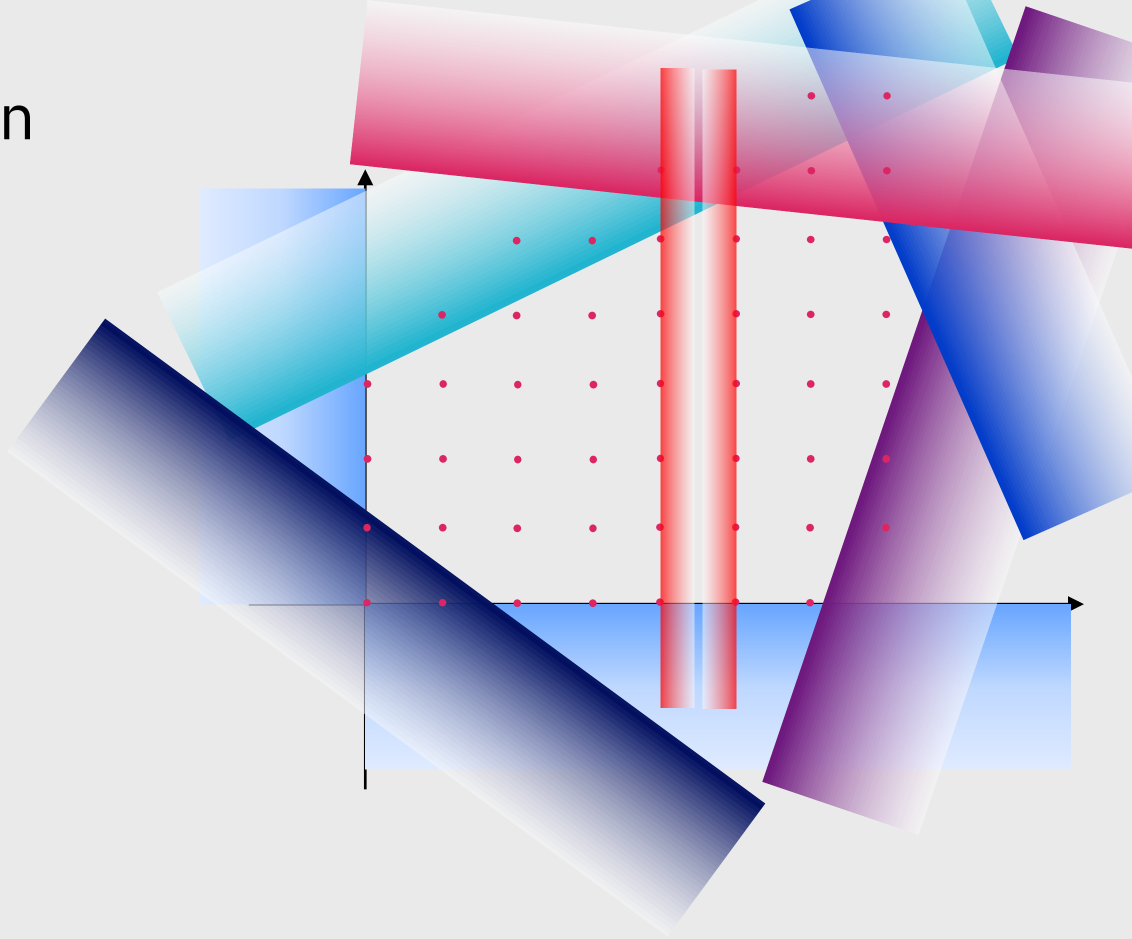
s.t.

$$Ax \geq b$$
$$x \geq 0$$

where

$$x \in \mathbb{Z}^n$$
$$A \in \mathbb{R}^{m*n}$$
$$c \in \mathbb{R}^n$$
$$b \in \mathbb{R}^m$$

Integer Linear Program



Academic context

Mathematical Optimization

$$\min c^T x$$

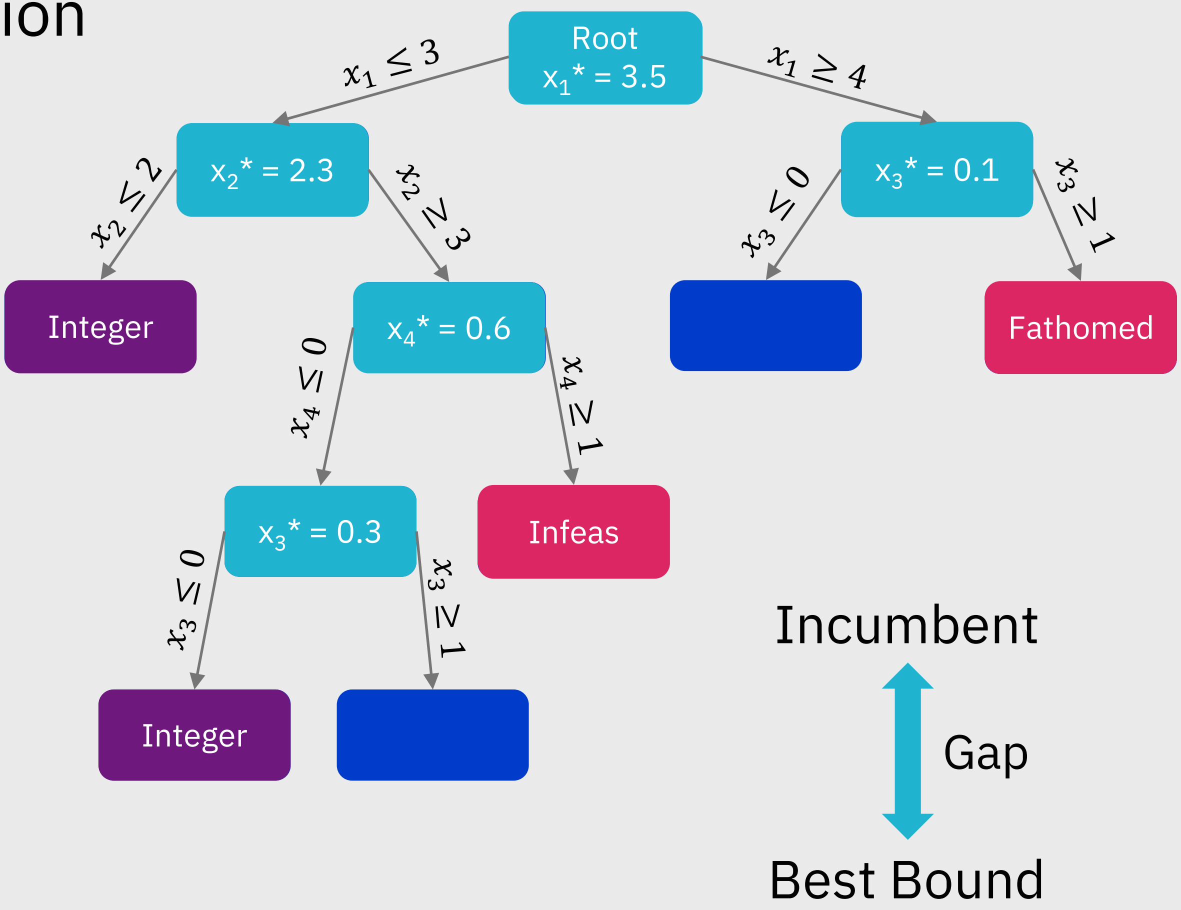
s.t.

$$Ax \geq b$$
$$x \geq 0$$

where

$$x \in \mathbb{Z}^n$$
$$A \in \mathbb{R}^{m*n}$$
$$c \in \mathbb{R}^n$$
$$b \in \mathbb{R}^m$$

Integer Linear Program

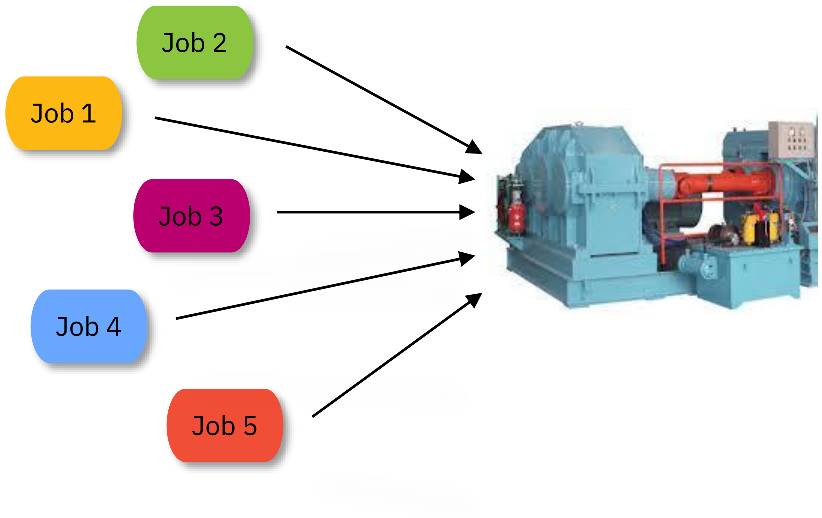


think

An Example

Sequencing

Optimization – an example



		To Job #				
		1	2	3	4	5
From Job #	1	--	8	4	5	3
	2	--	--	12	5	5
	3	--	9	--	7	3
	4	--	1	10	--	15
	5	--	7	13	3	--

Let’s look at an example of sequence dependent setups

Time shown is the downtime necessary to switch from one job to the next.

Optimization – an example: Schedules and Evaluation

	1st	2nd	3rd	4th	5th		Setup 1-2	Setup 2-3	Setup 3-4	Setup 4-5		Total Setup
Job #	1	2	3	4	5		8	12	7	15		42

First come, first serve

Optimization – an example: Schedules and Evaluation

	1st	2nd	3rd	4th	5th		Setup 1-2	Setup 2-3	Setup 3-4	Setup 4-5		Total Setup
Job #	1	2	3	4	5		8	12	7	15		42
	1	4	5	3	2		5	15	13	9		42
	1	2	4	5	3		8	5	15	13		41
	1	4	5	2	3		5	15	7	12		39
	1	2	5	3	4		8	5	13	7		33
	1	3	2	4	5		4	9	5	15		33
	1	3	4	5	2		4	7	15	7		33
	1	5	3	2	4		3	13	9	5		30
	1	4	3	2	5		5	10	9	5		29
	1	5	2	3	4		3	7	12	7		29
	1	2	3	5	4		8	12	3	3		26
	1	2	4	3	5		8	5	10	3		26
	1	2	5	4	3		8	5	3	10		26
	1	4	3	5	2		5	10	3	7		25
	1	5	2	4	3		3	7	5	10		25
	1	5	4	3	2		3	3	10	9		25
	1	4	2	5	3		5	1	5	13		24
	1	5	3	4	2		3	13	7	1		24
	1	3	2	5	4		4	9	5	3		21
	1	4	2	3	5		5	1	12	3		21
	1	3	5	2	4		4	3	7	5		19

First come, first serve

Rule based heuristics

Optimization – an example: Schedules and Evaluation

	1st	2nd	3rd	4th	5th	Setup 1-2	Setup 2-3	Setup 3-4	Setup 4-5	Total Setup	% optimal
Job #	1	2	3	4	5	8	12	7	15	42	382%
	1	4	5	3	2	5	15	13	9	42	382%
	1	2	4	5	3	8	5	15	13	41	373%
	1	4	5	2	3	5	15	7	12	39	355%
	1	2	5	3	4	8	5	13	7	33	300%
	1	3	2	4	5	4	9	5	15	33	300%
	1	3	4	5	2	4	7	15	7	33	300%
	1	5	3	2	4	3	13	9	5	30	273%
	1	4	3	2	5	5	10	9	5	29	264%
	1	5	2	3	4	3	7	12	7	29	264%
	1	2	3	5	4	8	12	3	3	26	236%
	1	2	4	3	5	8	5	10	3	26	236%
	1	2	5	4	3	8	5	3	10	26	236%
	1	4	3	5	2	5	10	3	7	25	227%
	1	5	2	4	3	3	7	5	10	25	227%
	1	5	4	3	2	3	3	10	9	25	227%
	1	4	2	5	3	5	1	5	13	24	218%
	1	5	3	4	2	3	13	7	1	24	218%
	1	3	2	5	4	4	9	5	3	21	191%
	1	4	2	3	5	5	1	12	3	21	191%
	1	3	5	2	4	4	3	7	5	19	173%
	1	5	4	2	3	3	3	1	12	19	173%
	1	3	4	2	5	4	7	1	5	17	155%
	1	3	5	4	2	4	3	3	1	11	100%

First come, first serve

Rule based heuristics

Optimal

Decision Optimization – an example

How did your solution do?

There are 24 possible combinations to schedule 5 jobs :

- 4 Options to chose from for the second job
- 3 for the third
- 2 to chose the fourth job
- The last job remaining gets to go fifth

number of combinations grows **very** fast

# Jobs	# combinations
3	2
4	6
5	24
6	120
7	720
8	5.040
9	40.320
10	362.880
11	3.628.800
12	39.916.800
13	479.001.600

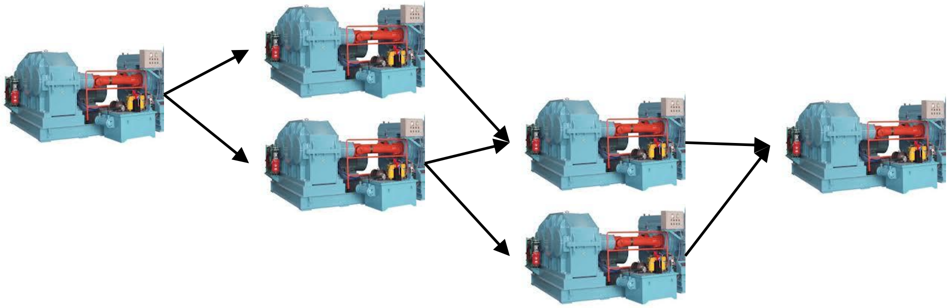
Decision Optimization – an example

Now, what if it‘s not just one machine?

And you need to consider alternatives?

Different process times on machines?

Resources that are replenished over time?



How do you decide the best sequence for ...

– the shortest makespan?

– the highest throughput?

– the least production cost?

– a combination of the above?

And what do you do if something changes?

Plans are nothing, Planning is everything.

– Dwight D. Eisenhower

What Is Decision Optimization?

- Manage resource efficiency, utilization and allocation
- Resources can be a number of things:

Resources	Choices to make
Capital	Invest, allocate
People	Hire, assign, schedule
Equipment	Acquire, schedule, locate
Facilities	Locate, size, schedule, maintain
Vehicles	Acquire, route, schedule, deliver, maintain
Material/Product	Acquire, allocate, produce, deliver, maintain

- Keywords to look out for:
  - Minimize and maximize
  - How many, how much, which, when, where
  - Decide, choose, plan, schedule, assign, route, source, maintain, locate, trade-off

think

Decision Optimization

vs

Machine Learning

- Learns “arbitrary function” from data
- Setup is a bit of an art
- Outcome is probabilistic
- Models can degrade over time
- Queries are usually fast
- Training process is time consuming*
- Impossible to explain and validate*

ML

- Takes description of a feasible solution and finds the optimal one
- Modelling is a bit of an art
- Outcome is deterministic*
- No Training
- Does not degrade over time
- Solve process is time consuming*
- Easy to validate, hard to explain

DO

An Example

Assignment Problems

Marketing Campaign Optimization

You are a marketing campaign planner – decide which offers to extend to which customers to maximize expected revenue

- **Input**
  – Predicted revenue per customer, per offer (output from a machine learning model)
- **Decisions**
  – Select up to one offer for each customer
- **Rules or Constraints**
  – Each offer can be used at most 3 times
- **Objective**
  – Maximize total expected revenue

Customers	Mortgage	Savings	Pension
1	70	60	60
2	80	80	70
3	90	90	80
4	50	50	50
5	100	100	100
6	110	130	150
7	20	20	90
8	10	40	80
9	0	50	60
10	40	40	80

Expected revenue of offer(“Savings”) to customer(“9”) = 50

Marketing Campaign Optimization

How Did You Do?

- **The best possible total revenue is 770**
- **Sequential rules would fail to find this solution**
  - For customer 6 we didn’t pick the highest scoring campaign
- **Consider how difficult this would be if there were**
  - Millions of customers
  - 10s of campaigns
  - Additional rules or constraints

Customers	Mortgage	Savings	Pension
1	70	60	60
2	80	80	70
3	90	90	80
4	50	50	50
5	100	100	100
6	110	130	150
7	20	20	90
8	10	40	80
9	0	50	60
10	40	40	80

Marketing Campaign Optimization

Input

James X

- Gender = male
- Age= 45
- Income=$200,000
- School-age children = 2
- Homeowner = yes
- Potential offers:
  - Mortgage
  - Savings
  - Retirement

Machine Learning model

Revenue scoring model

Output

A score (a number)

Predicted revenue score

Mortgage: 50
Savings: 100
Retirement: 130

Clients	Mortgage	Savings	Pension
1	70	60	60
2	80	80	70
3	90	90	80
4	50	50	50
5	100	100	100
6	110	130	150
7	20	20	90
8	10	40	80
9	0	50	60
10	40	40	80

Marketing Campaign Optimization

Input

- List of clients
- List of offers
- Predicted revenue per client, per offer
- Rules
  - At most 1 offer per client
  - Use each offer at most 3 times

Decision Optimization model

Campaign optimization model

Output

A plan across all clients & offers

Marketing campaign plan

Clients	Mortgage	Savings	Pension
1	70		
2		80	
3		90	
4	50		
5	100		
6		130	
7			90
8			80
9			
10			80

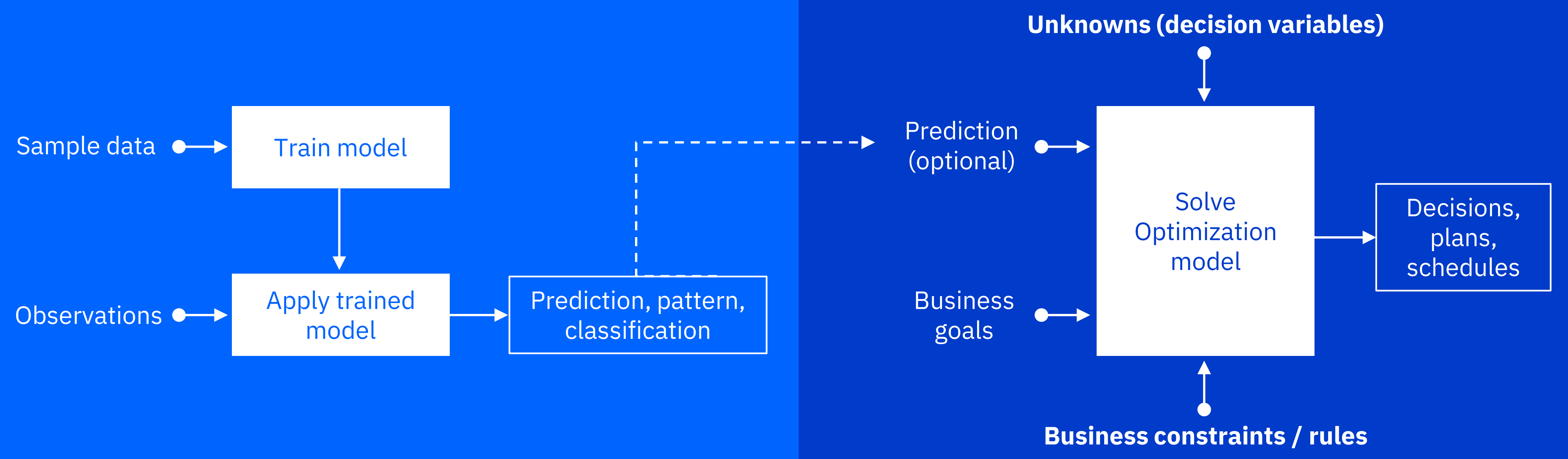
Machine learning and optimization: better together

Predictive analytics

- Basic (supervised): You **know the answer**, and you **train the machine how to find it**.
- Advanced: Unsupervised, reinforcement, deep learning

Prescriptive analytics

- You **don’t know the answer**, and you **provide the machine the logic on what is a good and a bad solution**.
- Advanced: Robust, stochastic, etc.



Tech

One Product – two Engines

CPLEX

- High performance (Mixed Integer) Linear & Quadratic Program solver
- Branch & Cut Framework
- Simplex & Barrier Methods
- Primal & Dual Heuristics, Cutting planes, ..
- Solution quality metric: MIP Gap
- Highly configurable, customizable and extensible

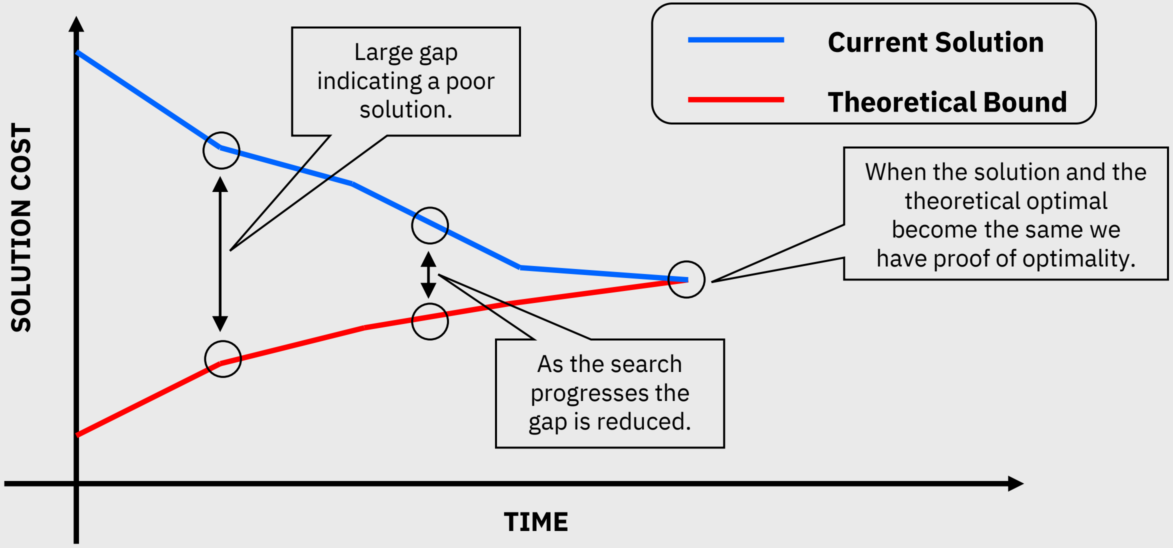
CP Optimizer

- Constraint Programming & Constraint-based Scheduling
- Particularly well suited for large scale scheduling of activities and resources
- Whenever the order of activities in time is relevant, e.g. precedence & synchronization constraints
- Resource usage and blocking
- State tracking & transition times

Optimization Engines – CPLEX – Key Concepts

Optimality Gap

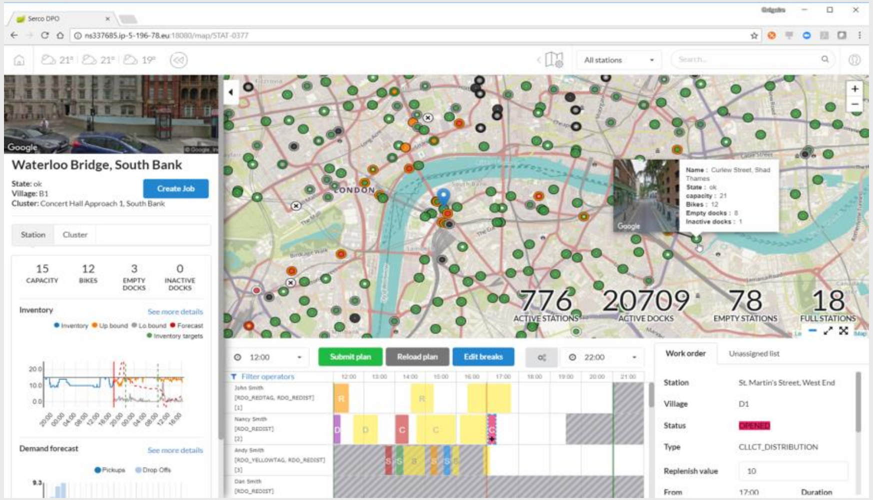
How far away is my current solution from the best possible solution?



Use Cases

LONDON Bike Sharing System

- Wholistic decision support for daily and strategic planning
- Predicting fluctuations in demand
- Enabling quick reaction to unforeseen events

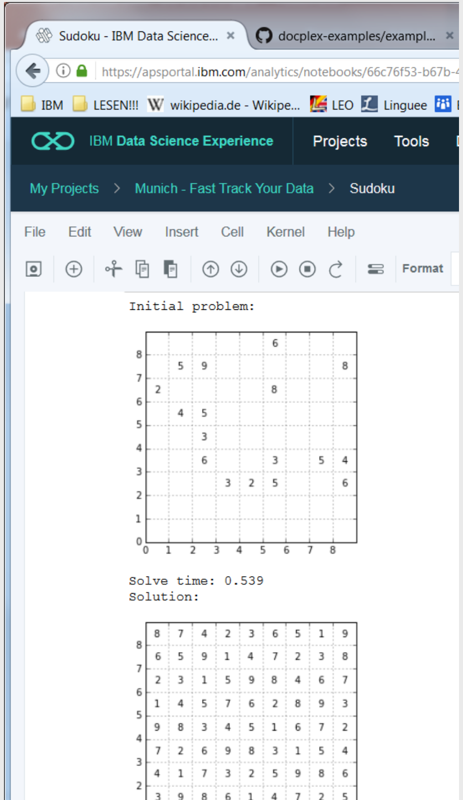


- Automating the nightly redistribution of bicycles
- Realtime-Monitoring (Stations, Vans)
- Optimized routing of vans
- Extension to plan collection of bikes requiring maintenance and abandoned bikes

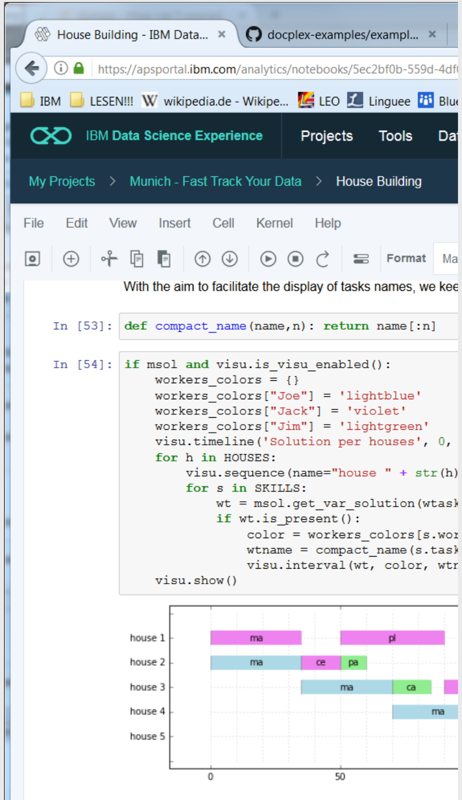
More examples

https://github.com/IBMDecisionOptimization/docplex-examples/tree/master/examples/cp/jupyter

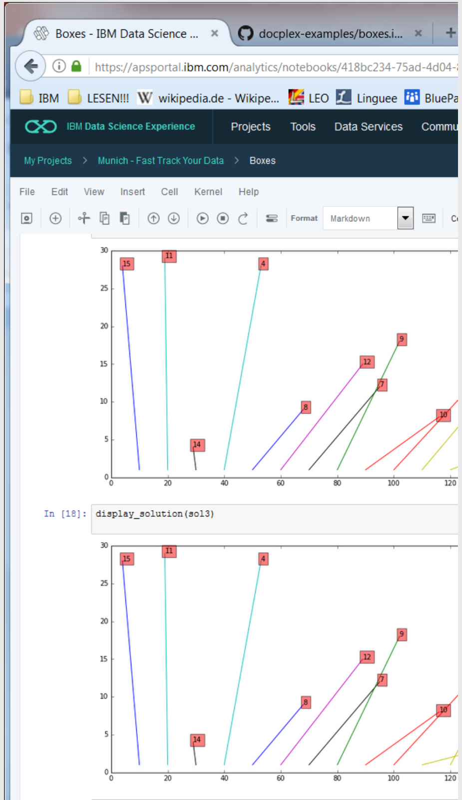
Sudoku



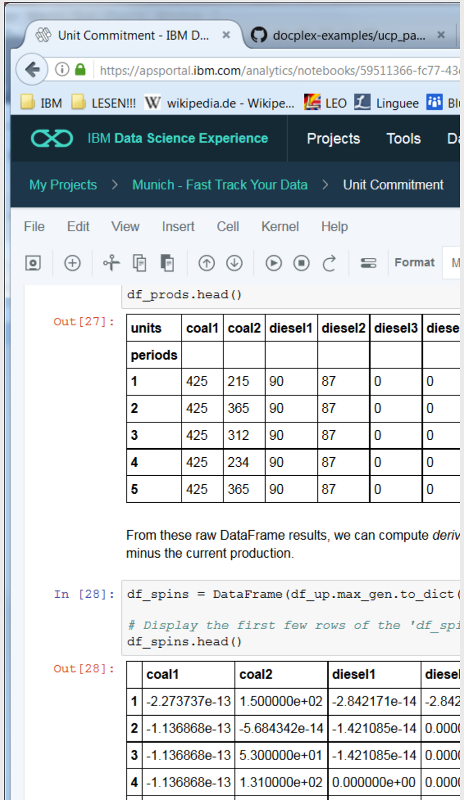
House Building



Box Placement



Unit Commitment



Thank you

Sebastian Fink
Client Technical Professional
Decision Optimization
IBM Analytics
—
sebastian.fink@de.ibm.com
+49 151 52 70 39 96
ibm.com

IBM®