UNIVERSITÄT
BIELEFELD
Faculty of Business Administration
and Economics

DECISION ANALYTICS

Combining OR and Data Science

Summer Term 2022

1.b Background and Motivation

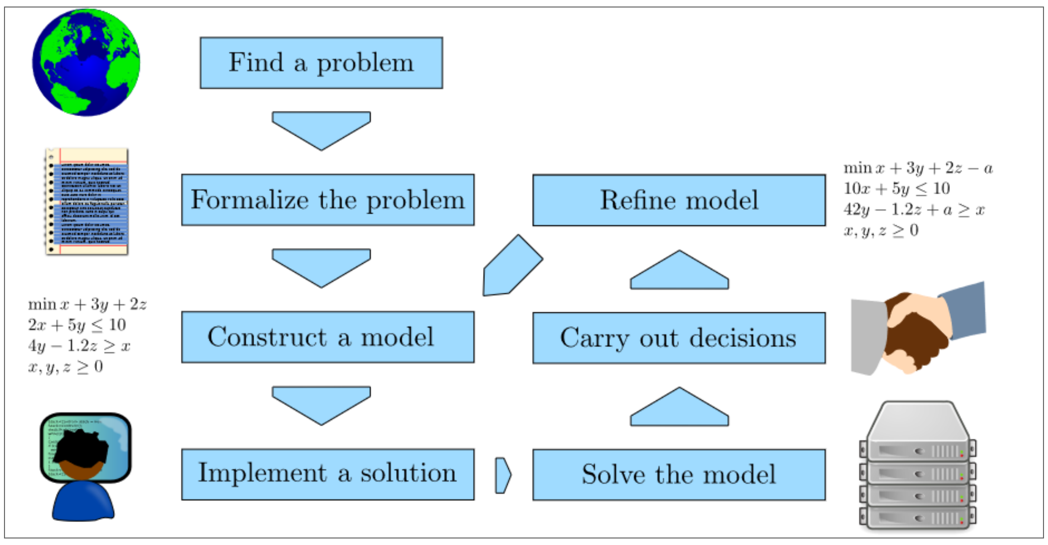
J-Prof. Dr. Michael Römer, Till Porrmann
Decision Analytics Group | Bielefeld University

Outline

- Operations Research and Data Science: A short review
- Combining OR and Data Science
- Examples for Combining OR and Data Science

Operations Research - What is that?

> **"Operations Research is the application of scientific and mathematical methods to the study and analysis of problems involving complex systems"** INFORMS (Institute for Operations Research and the Management Sciences)



Find a problem

Formalize the problem

Construct a model

Implement a solution

Solve the model

Carry out decisions

Refine model

$$\begin{aligned}&\min x + 3y + 2z\\&2x + 5y \leq 10\\&4y - 1.2z \geq x\\&x, y, z \geq 0\end{aligned}$$

$$\begin{aligned}&\min x + 3y + 2z - a\\&10x + 5y \leq 10\\&42y - 1.2z + a \geq x\\&x, y, z \geq 0\end{aligned}$$

Simple Example: Manufacturing Belts

Consider the following toy case study: Planning the daily production for small company that manufacture two types of belts: A and B

- The contribution margin is $2 for an A-belt and \$1.5 for a B-belt.
- The full production can be sold to a small chain of shops.
- Producing a belt of type A takes twice as long as producing one of type B, and the total time available in that day would allow producing 1000 belts of type B if only B-belts were produced.
- Both types of belts require the same amount of leather, and there is enough leather to produce 800 belts.
- The total number that can be produced per type is limited by the number of available bucks: The company has 400 bucks for type A and 700 bucks for type B.

Create an LP model that determines the number of belts from each type to produce if the shop aims at maximizing the total contribution margin!

Belt Manufacturing: LP Formulation

Set

- $I = \{A, B\}$ belt types

Decision Variables

- x_i: number of belts to produce from type i

$$\begin{aligned} \max\ & 2x_A + 1.5x_B \\ \text{s.t. } & 2x_A + x_B \leq 1000 \\ & x_A + x_B \leq 800 \\ & 0 \leq x_A \leq 400 \\ & 0 \leq x_B \leq 700 \end{aligned}$$

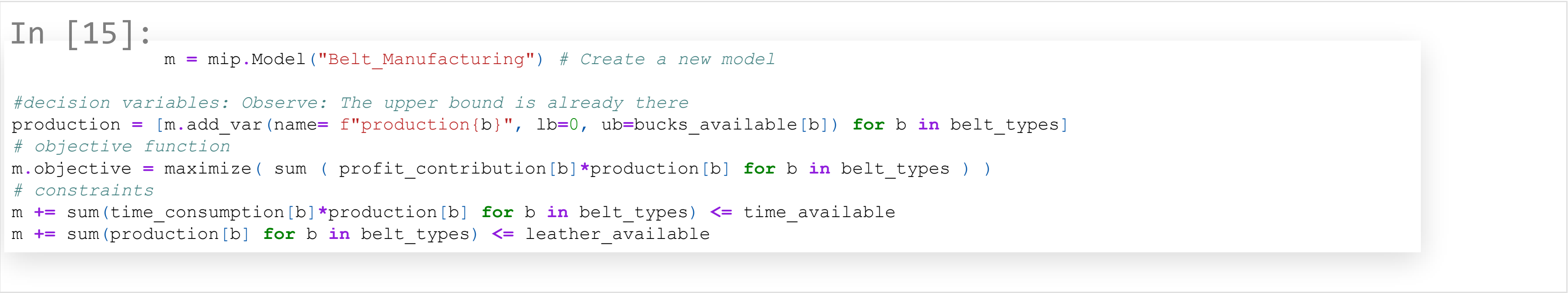
Belt Manufacturing: Python Implementation

- set up the model data:

```
In [14]:
                  #Data
belt_types = [0,1]
profit_contribution = [2, 1.5]
time_consumption = [2, 1]
time_available = 1000
leather_available = 800
bucks_available = [400, 700]
```

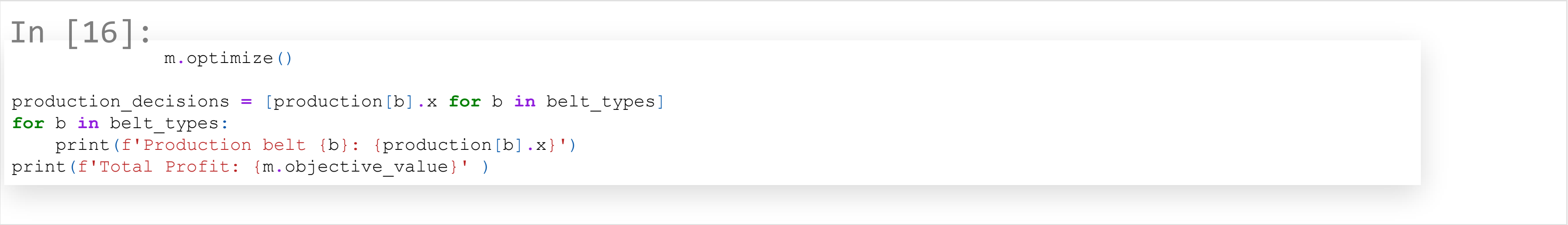
Belt Manufacturing: Python Implementation

- set up the model in Python-MIP:

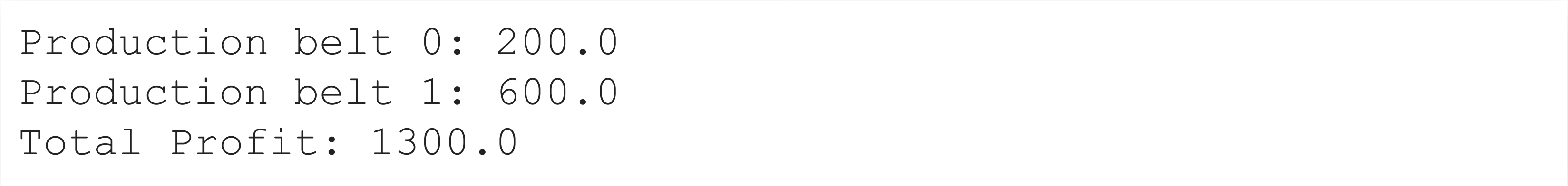


```
In [15]:
            m = mip.Model("Belt_Manufacturing") # Create a new model

#decision variables: Observe: The upper bound is already there
production = [m.add_var(name= f"production{b}", lb=0, ub=bucks_available[b]) for b in belt_types]
# objective function
m.objective = maximize( sum ( profit_contribution[b]*production[b] for b in belt_types ) )
# constraints
m += sum(time_consumption[b]*production[b] for b in belt_types) <= time_available
m += sum(production[b] for b in belt_types) <= leather_available
```

- solve the model and print results

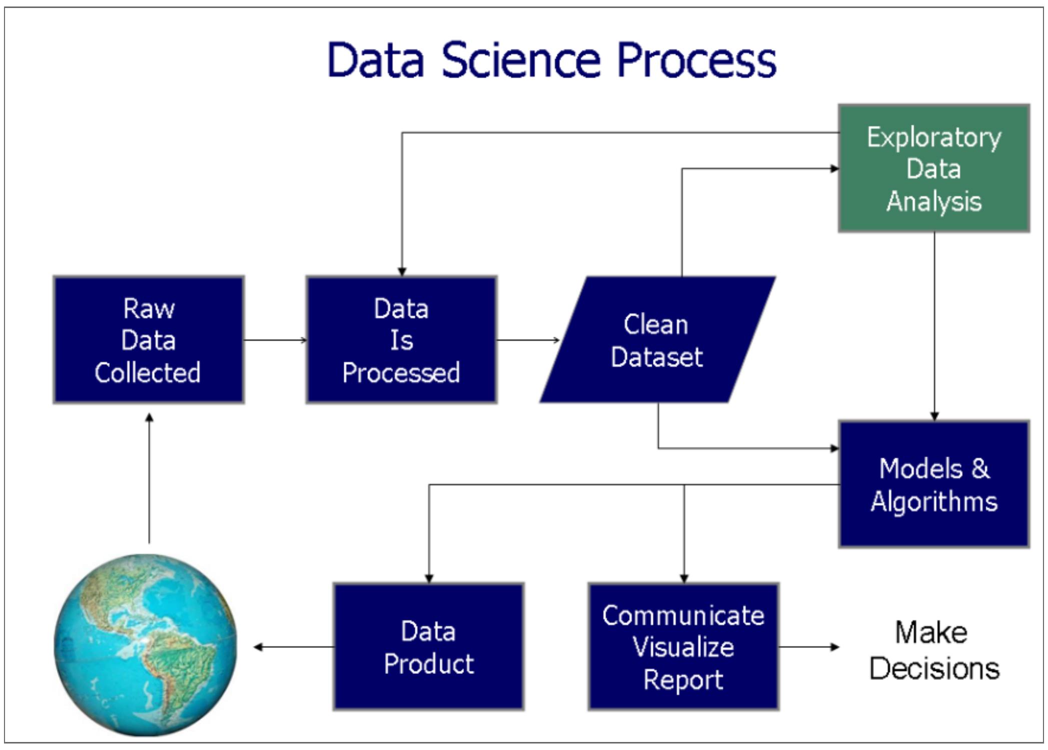


```
In [16]:
            m.optimize()

production_decisions = [production[b].x for b in belt_types]
for b in belt_types:
    print(f'Production belt {b}: {production[b].x}')
print(f'Total Profit: {m.objective_value}' )
```



```
Production belt 0: 200.0
Production belt 1: 600.0
Total Profit: 1300.0
```

Data Science



Source: "Doing Data Science by Schutt &O'Neil (2013)

Data Science: Some Relevant Terms

- **Machine learning** is a field of computer science that uses statis- tical techniques to give computer systems the ability to ”learn” (e.g., progressively improve performance on a specific task) with data, without being explicitly programmed – *Wikipedia*
- **Data mining** is the process of discovering patterns in large data sets involving methods at the intersection of machine learning, statistics, and database systems. – *Wikipedia*
- **Predictive modeling:** the process of developing a mathematical tool or model that generates an accurate prediction. – *Applied Predictive Modeling*
- **Predictive analytics** encompasses a variety of statistical techniques from data mining, predictive modelling, and machine learning, that analyze current and historical facts to make predictions about future or otherwise unknown events. - *Wikipedia*

An exact differentiation of the terms is not possible because many overlaps exist.

Machine Learning

- **Supervised learning:** A training set of examples with the correct responses (targets) is provided and, based on this training set, the algorithm generalizes to respond correctly to all possible inputs. This is also called learning from examples.
- **Unsupervised learning:** Correct responses are not provided, but instead the algorithm tries to identify similarities between the inputs so that inputs that have something in common are cat- egorized together. The statistical approach to unsupervised learning is known as density estimation.

Source: *Machine Learning: An Algorithmic Perspective*

Belt Example: Uncertain Machine Availability

The machine that is needed for producing the belts sometimes has failures that lead to reduced available machine time. Fortunately, the company has collected lots of data regarding the daily machine availability. The data contains has the following information for each day:

- Date, including weekday
- Temperature -Humidity
- Type of leather used for production
- Available time Question

What kind of ML task do we have here? How can Machine Learning support the production planning decision for the next day?

Data Analytics

> *"Analysis of data, also known as data analytics, is a process of inspecting, cleansing, transforming, and modeling data with the goal of discovering useful information, suggesting conclusions, and supporting decision-making" – Wikipedia*

What we will consider in this course:

- Aggregate and inspect data to help make operational decisions
- Influence our OR models with aggregated/transformed data
- Allow OR techniques to make data-driven decisions on their own 31-MM34 CORDS SoSe21: Back

Interfaces between OR and DS

- DS can feed OR models with parameters or even structural elements such as constraints
- DS techniques are often instrumental in algorithm development and during execution
- Data collection and machine learning for online / dynamic OR models
    - → Reinforcement Learning / Approximate Dynamic Programming
- Optimization (models) form(s) the basis for many data mining and machine learning methods

Belt Manufacturing Example: Predict and Optimize?

One obvious way for combining OR and Data Science in the belt manufacturing example:

- Train a regression model that predicts machine availability from data
- For a given planning problem (say for the next day) take all available information (e.g. weather forecast, leather type) and predict the expected availability
- Use that availability as a parameter in the optimization model

Is this the best way of dealing with the described problem?

Learning to answer this question is a key part of this course!

Some more Complex Examples

Example 1: Liner Shipping Service Design

- Paper from Kevin Tierney et al. (2019)
- **Goal:** Design liner shipping services with an **on-time-guarantee**

Example 1: Liner Shipping Service Design

Key Problem Feature: Uncertainty:

- The sailing time between two ports depends on varying environmental conditions and is thus an uncertain parameter

Naïve approach: Predict-and-Optimize

- Use a single point forecast (typically the expected value) for every uncertain parameter
- Solve a deterministic optimization model

→ Very common approach that often leads to suboptimal and non-robust solutions

Better approach: Account for uncertainty in the OR model

- Use DS techniques to quantify uncertainty, e.g. in form of probability distributions
- Feed this information into a stochastic optimization model

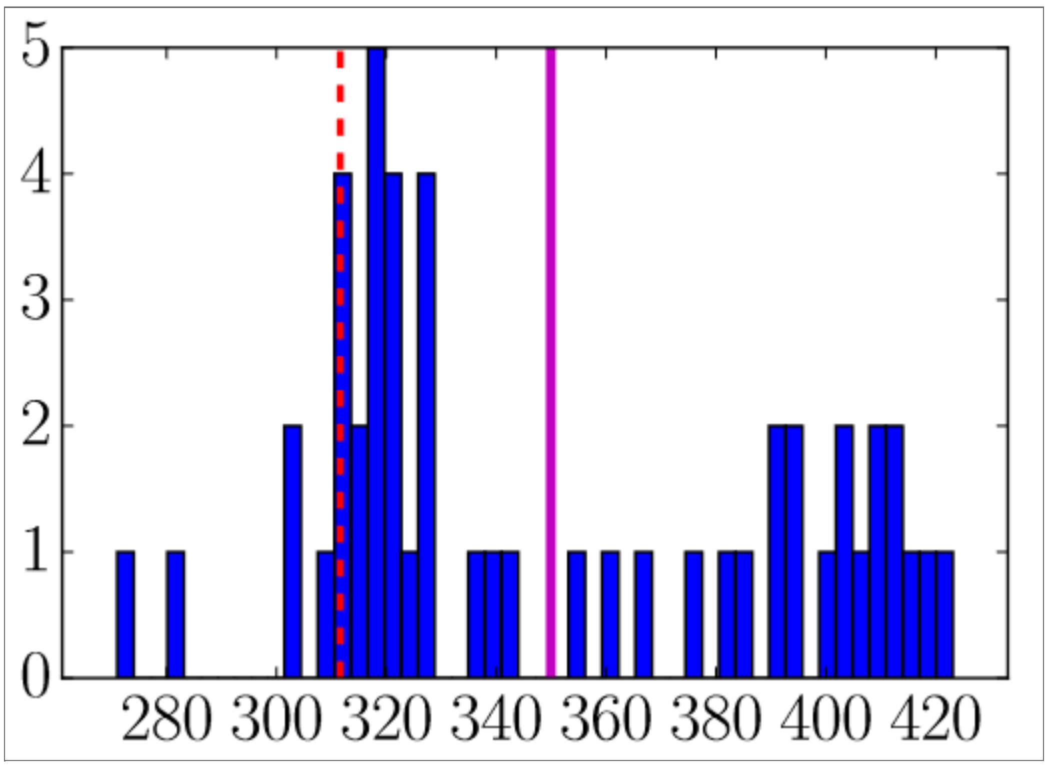
Example 1: Liner Shipping Service Design

Say we have an optimization model to design liner shipping routes. How can we guarantee that ships will be on time with a given probability?

The question, reformulated:

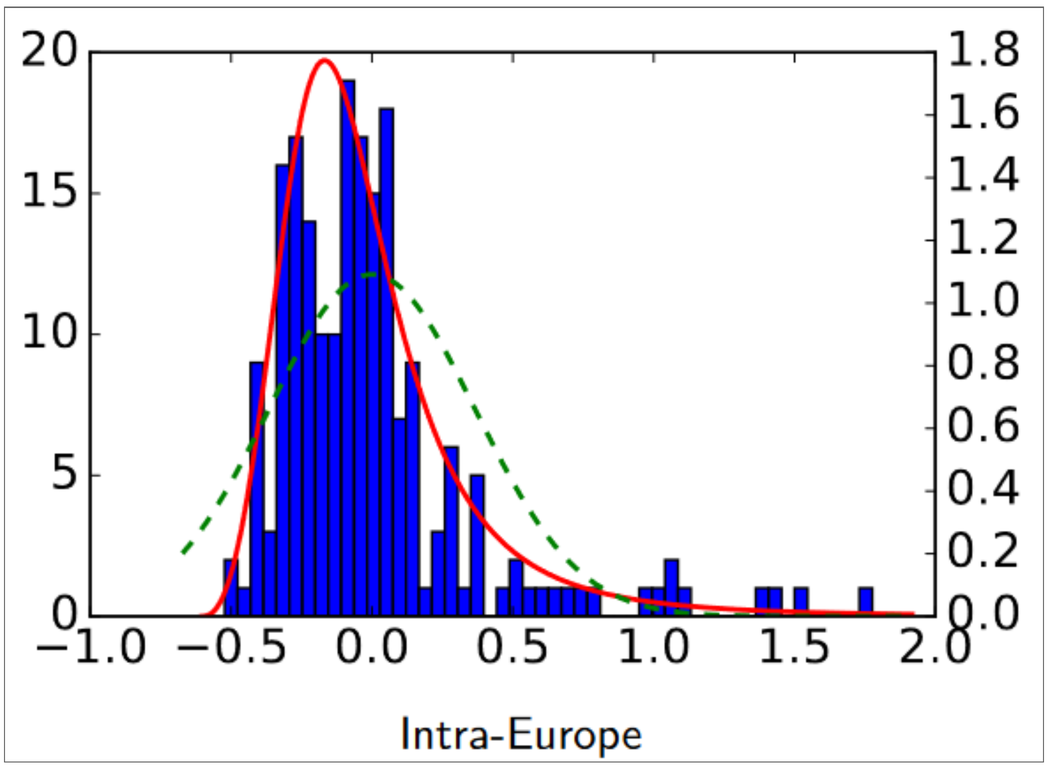
What distribution best represents the sailing time of container ships between ports? How can we use that distribution in an optimization model?

Travel Time Data: Single Connection



- empirical travel time from Singapore to Suez

Travel Time Data: Aggregation of Connections within Europe



- deviation from the mean travel time for each connection

And where’s the OR?

- These distributions obtained with DS techniques can now be used in an OR model for designing services by using a so-called chance constraint

$$x_i + t_{ij}^{\alpha} \leq x_j + M(1 - y_{ij})$$

Decision Variables and Parameters:

- x_i – Time at port i
- y_{ij} – Ship sails from i to j?
- t_{ij}^{α} – Minimum buffer needed for service level α

t_{ij}^{α} is the inverse CDF value at α of the given distributiom

Does it help?

Using a simulation, the services obtained with the chance constraint-model were compared to those obtained with a deterministic model. The key results are:

- Using the chance constraint model results in a big reduction in delays
- In order to achieve this, more vessels were needed
- By running the evaluation with different distributions and service levels, the decision makers can try finding their sweet spot in the trade-off between service level and cost

Example 2: Speeding Up CPLEX with Machine Learning

IBM CPLEX

- One of the leading commercial Mixed Integer Programming solvers
- In release 12.10, they significantly improved the Mixed Integer *Quadratic* Programming performance employing Machine Learning

Mixed Integer Quadratic Programming (MIQP)

- Considers Mixed Integer Programs with quadratic terms in the objective function
- In practice, harder to solve than Mixed Integer Linear Programming (MILP) problems

Source: P. Bonami, A. Lodi, G. Zarpellon (2019)

Example 2: Speeding Up CPLEX with Machine Learning

Binary quadratic terms can be addressed in two alternative ways:

- Linearize by applying a well-known reformulation technique and solve as linear problem -Do not linearize and directly solve the quadratic problem ..none of the two dominates the other

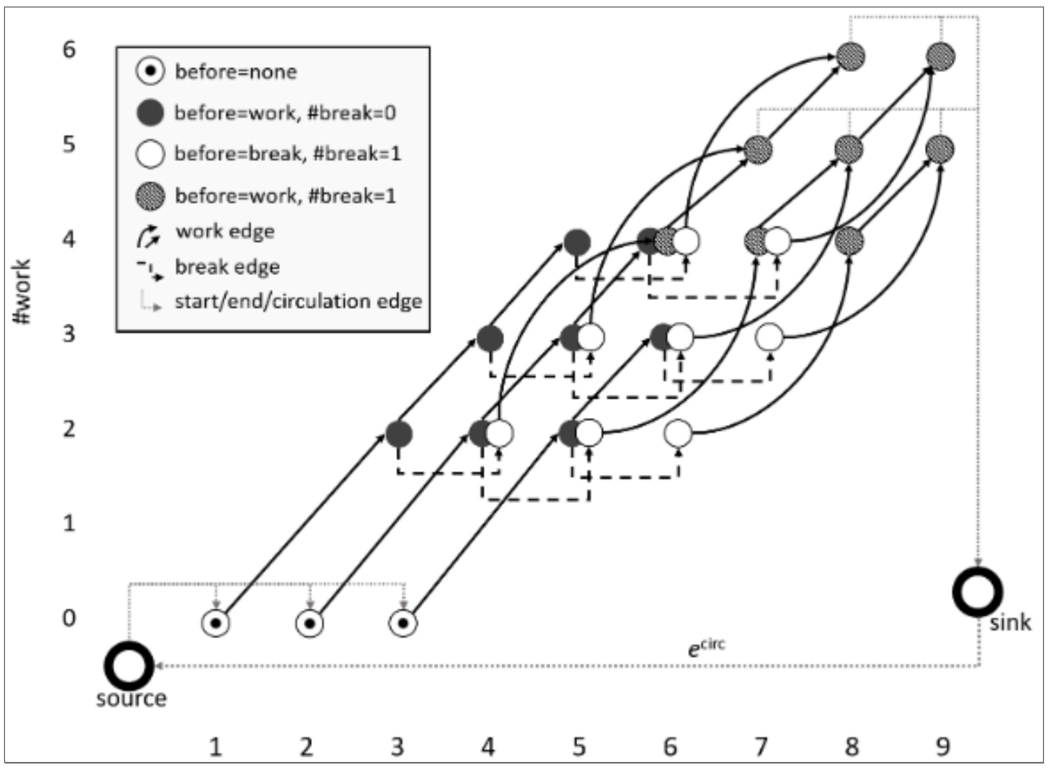
Idea: Train a classifier to predict the favorable approach

- SVM classifier based on 21 instance features and an estimation of the objective function difference of the two formulations
- The classifier has an accuracy of 79 %
- Average reduction of solution time of 49% on average for non-trivial instances

Example 3: Learning to Reduce MIP Models

State-Expanded Network Formulations for Personnel Scheduling

- in such a network, nodes are associated with states (e.g. number of hours worked), arcs are associated with assignments (e.g. a working from 2 to 4 pm) in a way that each path in the network corresponds to a feasible shift



Example 3: Learning to Reduce MIP Models

State-Expanded Network Formulations

- networks are used in MIP models as network flow components
- the models are very strong (there is a small LP-IP gap), but also very large

Idea: Train regression model that predicts which part of the network are important and which parts can be left out

- training with optimal solutions for different (but similar) instances
- substantial reduction of model size
- much faster solution time (only 15 % of the original time), only very small loss in solution quality (much less than 1 % on average)

Source: T. Porrmann and M. Römer (2021)

Conclusion

What we did today: We...

- Motivated topics of the course
- Investigated data analytics and decision making
- Learned something about ships, solving MIPQPs, and state-expanded networks

What’s next

- we may have a short (non-mandatory) tutorial on Python and Jupyter
- in two weeks we will start Part 1 of the course with the first block

References

Liner Shipping Example
Tierney K, Ehmke JF, Campbell AM, M¨uller D (2019) *Liner shipping single service design problem with arrival time service levels.* Flexible Services and Manufacturing Journal 31(3):620–652.

IBM CPLEX Example
Bonami P, Lodi A, Zarpellon G (2018) Learning a Classifi- cation of Mixed-Integer Quadratic Programming Problems. *Lecture Notes in Computer Science 10848*, Springer, 595–604.

Model Reduction Example
T. Porrmann and M. Römer (2021) Learning to Reduce State-Expanded Networks for Multi-Activity Shift Scheduling, *18th International Conference on the Integration of Constraint Programming, Artificial Intelligence and Operations Research (CPAIOR) 2021, Lecture Notes in Computer Science 12735*, Springer, 383-391.