**Context and objective**

**Problem:** An industrial goods distributor wants to find the optimal price that maximizes total revenues less transportation costs and storage costs.

**Solution:** Our solution effectively combines the use of Probabilistic Programming with MILP in a modular architecture that reflects the company value drivers’ tree:

1. **Identify the key drivers**
   - Endogenous and exogenous
   - Fat-tailed distributions

2. **Estimate distribution of key optimization parameters**
   - Generative modelling with Probabilistic Programming
   - Uncertainty in model and model parameters: Bayesian Machine Learning
   - MCMC and (Variational) Inference off-the-shelf

3. **Optimization**
   - Leverage Python interfaces to solvers (e.g. GurobiPy or Pyomo)
   - Robust and Stochastic Programming
   - Alternative methodological approaches: Meta-Heuristics, Reinforcement Learning, ...

**Key modules:**
- Estimation of the distribution of demand (for a product at a specific price point)
- Estimation of transportation and storage unit costs
- Mathematical Programming module that defines the optimal shipping strategy

**Conclusions**

- Generative models of key optimization parameters are necessary input to Robust Optimization and Stochastic Programming problems
- Ongoing work: seamless interoperability between Probabilistic Programming frameworks and Python interfaces to solvers