# ATM Replenishment Optimization Under Demand Uncertainty

Course: Introduction to Operational Research Techniques Using Data Science

Team Members: Denizcan Biçakçı – Hamza Alsatari – Metin Soydeğer

Date: May 2025

## 1. Introduction

ATM cash replenishment is a complex operational challenge faced by banks to ensure that customers have constant access to cash while also minimizing the cost of transporting and storing it. The problem lies in the uncertainty of customer withdrawal demand, which can vary significantly across locations and time periods. If a bank overestimates demand, they incur excess cash handling and storage costs. If they underestimate it, ATMs may run out of cash, leading to customer dissatisfaction and potential financial penalties.

In this project, we implemented the model described by Ekinci, Serban, and Duman (2021), which addresses this problem using a robust optimization approach. Unlike traditional models that use point forecasts, this method uses prediction intervals and minimizes the worst-case cost under demand uncertainty. Our objective was to replicate this model in Python and analyze its sensitivity to different parameters involved in the optimization process.

## 2. Problem Description and Model Overview

The goal of the ATM replenishment model is to determine how much cash to load into each ATM in each period (week) to minimize the total cost. The cost has two components: the cost of loading cash and the penalty incurred if actual customer demand exceeds the cash loaded. Since future demand is uncertain, the model does not rely on a single forecasted value but instead uses a forecast interval, derived from historical data.

The mathematical formulation is as follows:

min_x [ c * x + max_{d ∈ [L, U]} g * (d - x)^+ ]

Where:

x = decision variable, amount of cash to load
c = unit cost of loading cash
g = dissatisfaction penalty per unit of unmet demand
H = higher penalty when ATM is completely out of cash
[L, U] = forecast interval (lower and upper bounds of expected demand)

The objective function calculates the cost of loading cash plus the worst-case penalty over all possible demands within the interval [L, U]. This makes it a robust optimization problem that aims to make safe and cost-effective decisions under uncertainty. It is particularly useful in operational contexts where demand forecasts are imprecise or volatile.

## 3. Python Implementation Summary

The Python implementation is structured to mirror the robust optimization model proposed in the paper. The notebook loads ATM demand data, applies forecasting using linear regression, constructs prediction intervals, and then computes the optimal cash load using robust optimization.

Step-by-step overview of the code logic:
1. Data Loading and Preprocessing: Historical withdrawal data is imported and detrended to reduce noise from seasonality and weekly patterns.
2. Forecasting: A linear regression model fits the historical pattern and predicts the next 4 weeks' demand.
3. Prediction Interval: A 95% prediction interval is constructed around the point estimate to account for demand uncertainty.
4. Robust Optimization: The optimization model uses this interval to compute the optimal cash load for each ATM.
5. Scenario Testing: A sample of 9 ATMs is selected for simulation and testing over the 4-week horizon.

## 4. Forecasting Methodology & Data Splitting Clarification

During the presentation, a question was raised about whether we split the data into training and testing sets. This is a valid question typically associated with machine learning and prediction-based models.

Why we did not split the data:
The paper we implemented does not treat forecasting as a standalone prediction task. Instead, it uses the forecasted interval directly as input for a decision-making model. As such, the model prioritizes robust decisions over predictive accuracy. The goal is not to minimize prediction error, but to make safe, cost-effective replenishment decisions within the uncertainty range.

If we had split the data:
- Advantages:
  • Allows for validation of forecasting performance (e.g., RMSE, MAE)
  • Helps detect overfitting or poor generalization
- Disadvantages:
  • Reduces the amount of historical data available to estimate accurate intervals
  • Shifts focus away from robust optimization, potentially harming the quality of replenishment decisions

Therefore, we deliberately followed the paper's methodology but applied it on a selected sample of 9 ATMs to generate the prediction intervals that drive the optimization model.

## 5. Sensitivity Analysis

Sensitivity analysis explores how the model responds to changes in key parameters. We analyze the behavior of the optimal load decision under changes to:
- c (unit cost of loading cash)
- g (dissatisfaction penalty for unmet demand)
- H (high dissatisfaction penalty when ATM runs out of cash entirely)
- L, U (bounds of the demand prediction interval)

- x (decision variable: cash loaded)

Impact of increasing c:
- Higher loading costs discourage overloading. The model will tend to reduce the cash loaded to avoid excess cost.
- This may increase the risk of incurring dissatisfaction penalties if demand exceeds the load.

Impact of increasing g:
- As the penalty for unmet demand rises, the model becomes more conservative. It will increase the load amount to ensure coverage of worst-case demand.

Impact of increasing H:
- H reflects the cost when an ATM is completely empty and a customer cannot withdraw anything. This cost is typically higher than g.
- Increasing H drives the model to be even more conservative, potentially loading higher amounts to prevent this high-penalty outcome.

Impact of widening the interval [L, U]:
- A wider interval implies greater uncertainty. The model will load more cash to cover worst-case outcomes.

Impact of demand distribution shift:
- If average demand increases, both L and U shift up, causing an increase in the optimal load.
- Conversely, lower average demand reduces the load.

This analysis confirms that the model is sensitive and responsive to parameter changes, making it suitable for a dynamic and uncertain environment.

## 6. Model Limitations & Assumptions

While the model performs well in theory and simulations, it has several limitations and assumptions:
- No vehicle routing or delivery logistics: The model assumes cash can be delivered independently to each ATM.
- Demand falls within the prediction interval: If actual demand lies outside the forecast range, penalties may be underestimated.
- Fixed parameters: The costs (c, g, H) are constant across time and locations.
- Weekly re-optimization: The model operates on a weekly basis without intra-week adjustments.
- No adaptive learning: The model does not dynamically learn from past forecast errors or performance.

## 7. Conclusion

This project successfully replicated a robust optimization model for ATM cash replenishment as described in the academic literature. The approach integrates prediction intervals with optimization, allowing for better decision-making under demand uncertainty.

Our implementation followed the paper’s methodology closely, applying it to a 9-ATM sample and focusing on minimizing worst-case regret rather than improving prediction accuracy. We explored the logic, structure, and sensitivity of the model using a real-world inspired dataset and highlighted its strengths and limitations.

Robust optimization provides a meaningful improvement over traditional fixed-safety-stock strategies by being cautious in the face of uncertainty. Though the cost savings per ATM might be small, these benefits scale across a network of hundreds or thousands of ATMs, making this model practically valuable for banks.