Algorithmics: Project

Martin Stuwe, 676421

January 22, 2024

1 Introduction

The Nordic Film Days is an annual event that attracts movie enthusiasts from around the world. To make the most of this festival, we faced a challenging problem: how to create a schedule that allows us to watch as many different films as possible, optimize our viewing experience by selecting high-rated films, and minimize travel time between venues. In this report, we present our approach and implementation of this scheduling problem.

2 Problem Description

The Nordic Film Days involve simultaneous screenings of various films at different venues. Our goal is to create a schedule that adheres to the following criteria:

1. Maximize the number of different films watched (no duplicates).
2. Optimize the overall film rating based on IMDb rankings.
3. Minimize travel time between venues, considering delays caused by location changes and a one-minute grace period for joining screenings.

We are provided with data on films, including their title, weekday, start time, duration, location, and rating. Additionally, we have information about travel times between venues, as shown in Table 1.

3 Methodology

To solve this scheduling problem, we implemented a Python program using the PuLP package. The key steps in our methodology include:

1. Data preprocessing: Reading film data from CSV files and organizing it into appropriate data structures.

2. Linear programming formulation: Defining decision variables, the objective function, and constraints to maximize film ratings while avoiding overlaps and considering travel times.

3. Solving the linear program: Using PuLP to find the optimal film schedule that satisfies all constraints.

We maximize the summation over all showings multiplied by their respective rating. Subject to the contraints, that each movie can only be watched once and that overlapping movies cannot be watched. Our preprocessing steps in our python script determine with the help of a mapping of cinema tuples for the distances, which movies overlap to formulate the respective constraints. The integer program for one of the small datasets is:

$$
\begin{aligned}
&\text{MAXIMIZE } 5.9 \cdot \text{Showing}_0 + 5.9 \cdot \text{Showing}_1 + 5.8 \cdot \text{Showing}_{10} + \\
&3.8 \cdot \text{Showing}_{11} + 5.9 \cdot \text{Showing}_{12} + \\
&5.7 \cdot \text{Showing}_{13} + 5.9 \cdot \text{Showing}_{14} + \\
&5.9 \cdot \text{Showing}_2 + 5.8 \cdot \text{Showing}_3 + 5.9 \cdot \text{Showing}_4 + \\
&5.8 \cdot \text{Showing}_5 + 5.9 \cdot \text{Showing}_6 + 5.9 \cdot \text{Showing}_7 + 4.3 \cdot \text{Showing}_8 + 5.9 \cdot \text{Showing}_9 + 0.0 \\
\text{SUBJECT TO} \\
&\text{Showing}_7 + \text{Showing}_8 \leq 1 \\
&\text{Showing}_3 + \text{Showing}_5 \leq 1 \\
&\text{Showing}_6 \leq 1 \\
&\text{Showing}_8 \leq 1 \\
&\text{Showing}_2 + \text{Showing}_7 \leq 1 \\
&\text{Showing}_0 + \text{Showing}_{14} + \text{Showing}_4 + \text{Showing}_9 \leq 1 \\
&\text{Showing}_1 + \text{Showing}_{12} \leq 1 \\
&\text{Showing}_{13} \leq 1 \\
&\text{Showing}_{11} \leq 1 \\
&\text{Showing}_{10} \leq 1 \\
&\text{Showing}_i \in \{0, 1\} \quad \forall i \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14\}
\end{aligned}
$$

We use binary variables to decide whether we watch a movie or not. To avoid watching the same movie, we sum the respective variables for the same films at different locations/times and require that the sum be less than or equal to 1. The same logic applies to overlapping movies.

4 Implementation

Our Python implementation effectively handles the scheduling problem for multiple datasets. We create LP variables, define the objective function to maximize

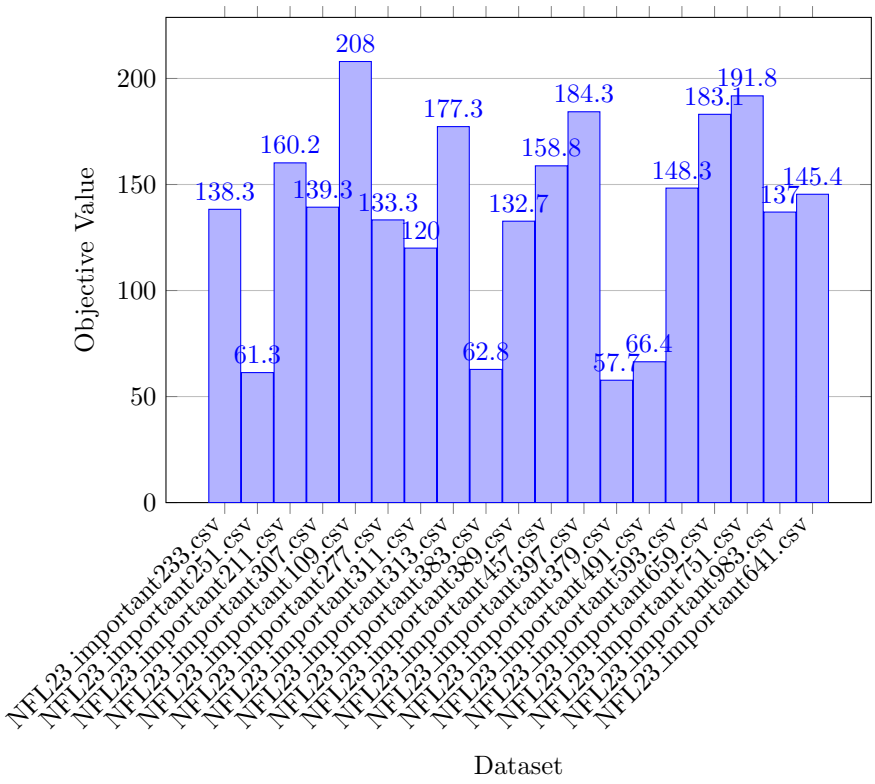

Figure 1: Objective Values for Different Datasets

the overall rating, and set up constraints to ensure no overlaps and no duplicates. We also consider travel times and delays when constructing the schedule. Our implementation generates output files containing the results for each dataset. The implementation can be found in main.py.

5 Results

We successfully solved the scheduling problem for various datasets, providing optimal movie schedules that maximize IMDb ratings. We recorded the objective values and runtime for each dataset. The results are stored in output files for further analysis.

The respective runtimes are:

Dataset, Objective Value, Runtime (seconds)
NFL23_important233.csv, 138.29999999999998, 0.16588878631591797
NFL23_important251.csv, 61.3, 0.9919431209564209
NFL23_important211.csv, 160.20000000000005, 0.2692911624908447
NFL23_important307.csv, 139.29999999999998, 0.16666126251220703

NFL23_important109.csv, 208.00000000000003, 0.4543285369873047
NFL23_important277.csv, 133.29999999999998, 0.15290594100952148
NFL23_important311.csv, 120.0, 0.12831354141235352
NFL23_important313.csv, 177.30000000000004, 0.3689141273498535
NFL23_important383.csv, 62.8, 0.9265275001525879
NFL23_important389.csv, 132.7, 0.09563016891479492
NFL23_important457.csv, 158.8, 0.19332408905029297
NFL23_important397.csv, 184.29999999999995, 0.39622068405151367
NFL23_important379.csv, 57.699999999999996, 0.239091157913208
NFL23_important491.csv, 66.39999999999999, 1.3689641952514648
NFL23_important593.csv, 148.3, 0.38736581802368164
NFL23_important659.csv, 183.10000000000002, 0.2737290859222412
NFL23_important751.csv, 191.80000000000004, 0.31435537338256836
NFL23_important983.csv, 137.0, 0.17076683044433594
NFL23_important641.csv, 145.4, 0.21124863624572754

6 Conclusion

In conclusion, our project for which we spend 10 hours tackles the scheduling problem for optimizing film festival attendance. We have implemented an efficient solution using linear programming techniques, Python, and the PuLP package. Our approach successfully meets the criteria of maximizing diverse film selection and optimizing overall rating while considering travel times. The results obtained demonstrate the effectiveness of our solution in creating optimal film schedules for the Nordic Film Days.