MIP Optimization: Fixed Charge Production Problem

Dunovan Rodgers

GB730 Final Project

AI Prompting for Problem Examples

- When asked to provide some examples of MIP optimization models, Google Gemini provided several options

The "Fixed Charge" Production Problem

The Concept: This is the classic introduction to MIP. In Linear Programming, costs are usually linear (make 1 unit = $5, make 2 units = $10). In the real world, you often have to pay a **setup fee** just to turn the machine on.

- **Scenario:** You manufacture two products (Widgets and Gadgets).
- **Binary Decision (y):** Should you turn on the Widget machine? Should you turn on the Gadget machine? (0 = No, 1 = Yes).
- **Continuous Decision (x):** How many Widgets and Gadgets should you produce?
- **Objective:** Maximize Profit (Revenue - Variable Costs - **Fixed Setup Costs**).
- **Key Constraint (The "Big M"):**
  - You cannot produce items if the machine is off.
  - $x_{widget} \leq M \cdot y_{widget}$ (Where M is a large number, representing the max capacity).
  - *Logic:* If $y = 0$, then x must be 0. If $y = 1$, x can be anything up to M.

Expanding on the Problem

- Gemini was asked to apply that concept to a more complex, realistic scenario with additional decision variables and constraints
- The Scenario: **Morning Grind Roastery**
- 3 Different Coffee Blends
- 9 Parameters
- 6 Decision Variables
- 6 Unique Constraints
- 1 Objective: Maximize Total Profit

Decision Variables

- Continuous variables (3): Number of pounds of each coffee blend to produce
- Binary variables (3): Yes/No decision of whether to produce each blend or not
  - There is a fixed setup cost associated with preparing the machines for each different blend

Constraints

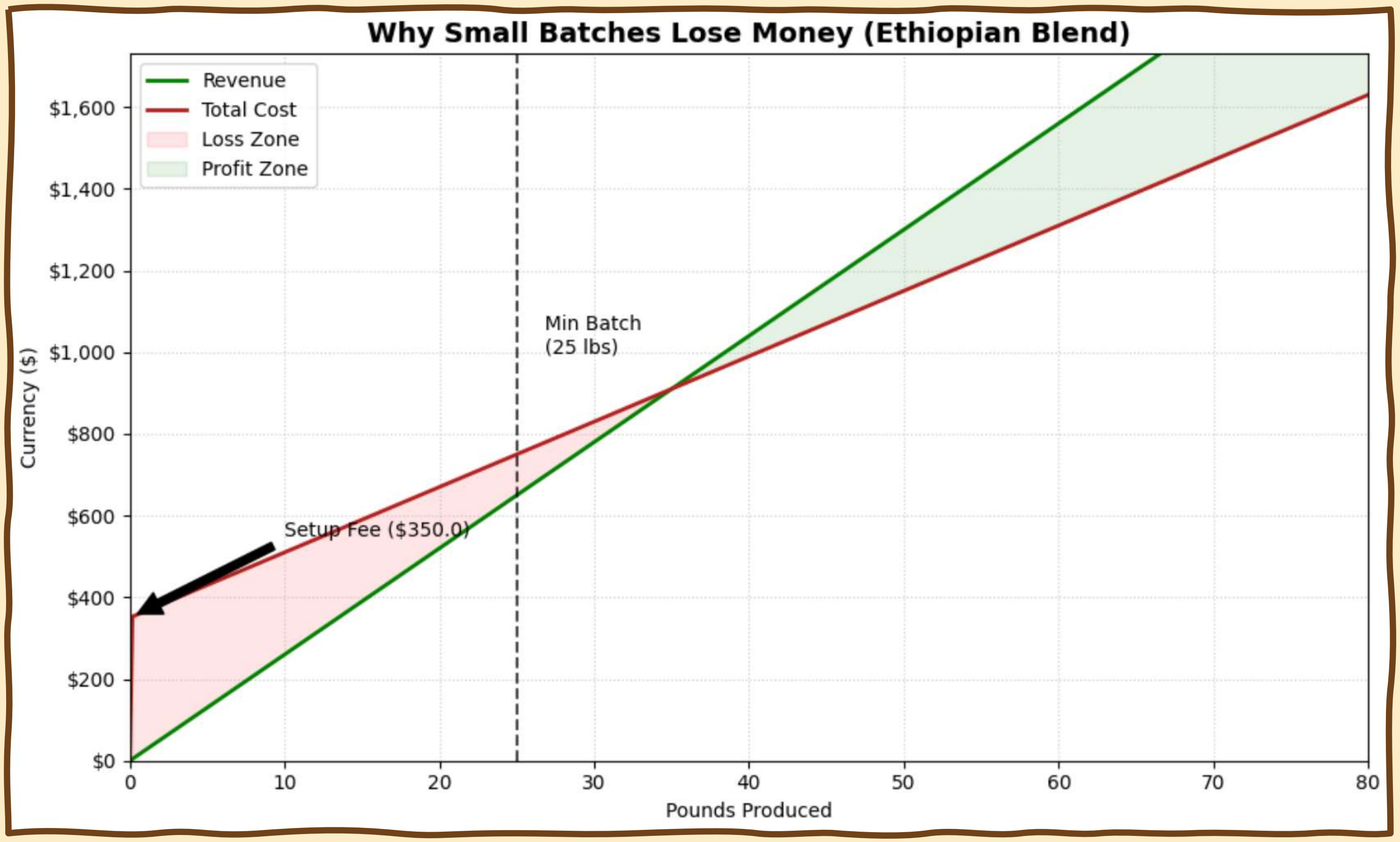
- Minimum batch size – If a batch is made, it can't be tiny
- Machine time – Machines can only run 40 hours weekly
- Warehouse volume – We only have 250 cubic feet of storage
- Variety – Must produce at least two different blends
- Demand – Cannot produce more than expected demand per batch
- "Big M" / Integrality – Blend variable is binary, batch size must be positive

Model Results

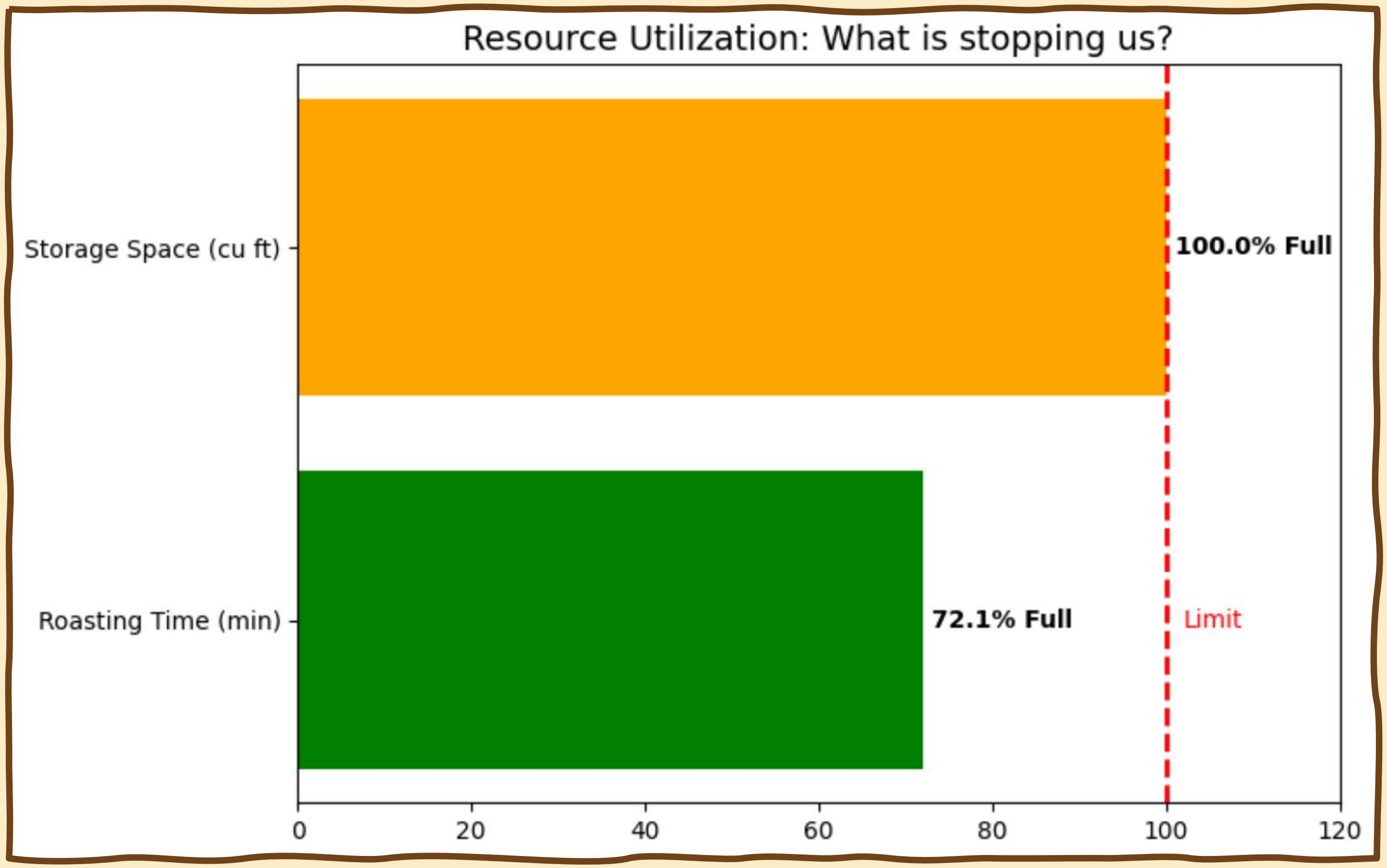
- All three blends created
- Storage maxed out
- Only 29 hours of machine hours utilized

```
Total Profit: $18,930.00
------------------------------
Blend            | Made? | Amount (lbs)
------------------------------
House            | Yes   | 400.0
Espresso         | Yes   | 1500.0
Ethiopian        | Yes   | 600.0
------------------------------
Time Used:    1730.0 / 2400 min
Storage Used: 250.0 / 250 cu ft
```

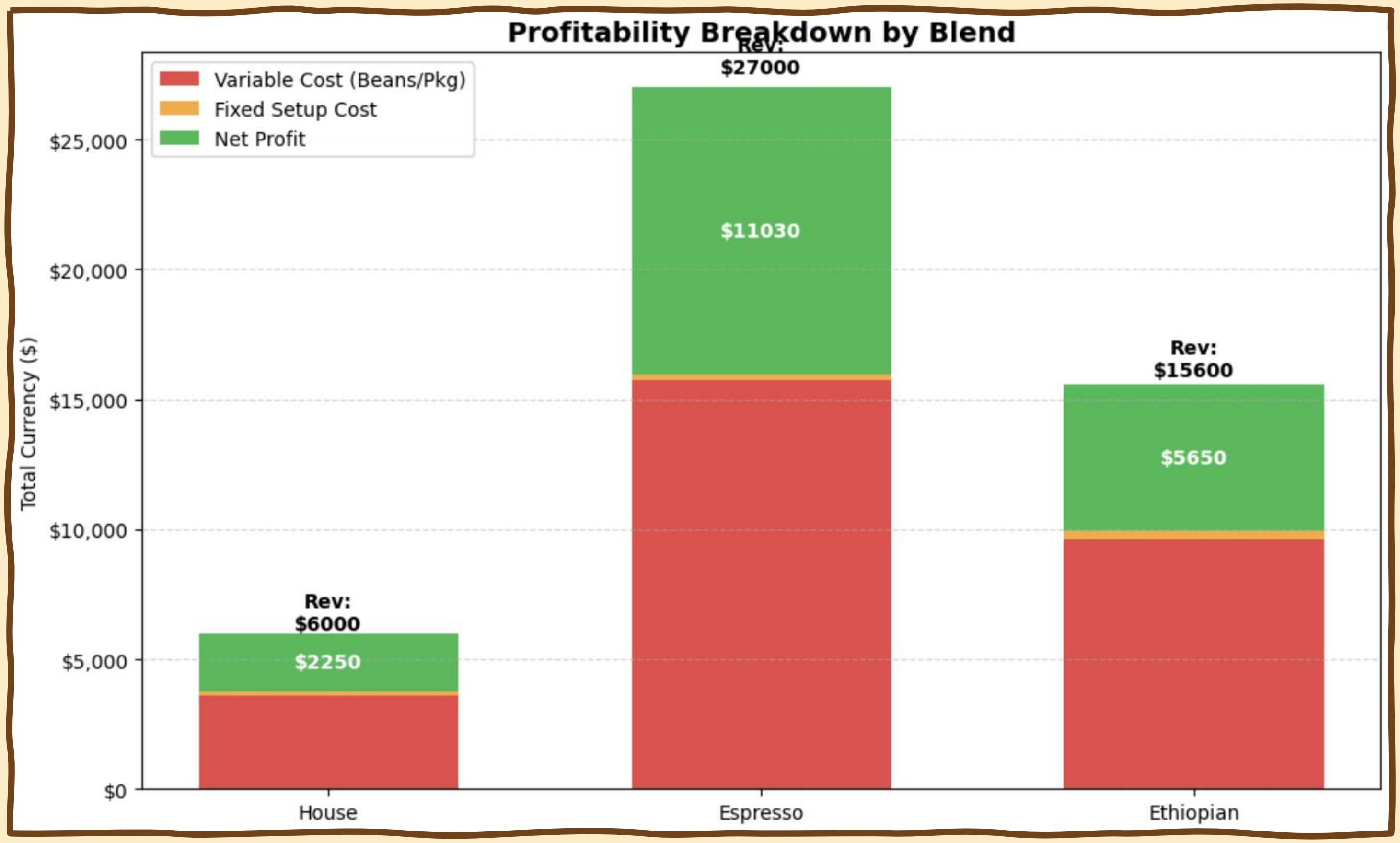
Break Even & Minimum Batch



Total Resource Utilization



Stacked Margin Analysis



Discussion Points

- 99% of code was AI written. I only made minor fixes
- Initial visualizations by the AI were poor. After asking it to try again, the quality and insight was incredible
- For a small roastery, this may be somewhat applicable, but a large operation would have much more complex restraints and far more parameters
- This model doesn't factor in fluctuations in demand, inventory costs for unsold goods, or volatility in materials cost

Insight & Recommendation

- The current storage constraints are resulting in unused machine time
- Increasing the storage to at least 400 cubic feet shifts the bottleneck to the 40 allotted machine hours
- This additional space would increase revenues by **43% -> $27,137**