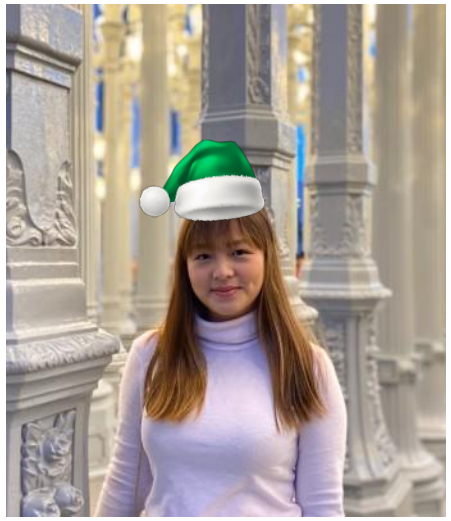
Santa's Workshop Schedule Optimization

ADSP 32013 Team 10
Bianca Gunawan, Jacky Lin, Sally Lee, Kevin Ko

Our Festive Crew



Bianca Gunawan

Santa’s Project Magic Elf

Jacky Lin

Santa's Data Wizard Elf

Kevin Ko

Santa's Data Wrangler Elf

Sally Lee

Santa's Data Sleuth Elf

Agenda


Business Problem
Data Overview
Problem Approach
Testing & Validation
Results & Limitations

Business Problem

Optimize Santa's Workshop Schedule to Minimize Cost

- Santa Claus is inviting **5,000 families** to come visit his workshop over **100 days**
- Each family has provided a **list of preferred dates** they would like to attend the workshop
- Due to high demand, it's not possible for everyone to get their top choices
- Santa aims to optimize the scheduling process to ensure fairness and **minimize any additional expenses** that may arise from accommodating families' preferences

Data Overview

Data contains families' preferred days to visit and the number of people in each family

Data Definition	Data Source	Data Size	Rows/Column
Santa's Workshop Tour	Github Data	177 KB	(5000, 12)

	family_id	choice_0	choice_1	choice_2	choice_3	choice_4	choice_5	choice_6	choice_7	choice_8	choice_9	n_people
0	0	52	38	12	82	33	75	64	76	10	28	4
1	1	26	4	82	5	11	47	38	6	66	61	4
2	2	100	54	25	12	27	82	10	89	80	33	3
3	3	2	95	1	96	32	6	40	31	9	59	2
4	4	53	1	47	93	26	3	46	16	42	39	4

- family_id: unique identifier for each family
- choice_0, choice_1, ..., choice_9: each family's first to last preferred day to visit Santa Workshop
- n_people: number of people in each family

Problem Approach

Methods - Integer Programming

- **Binary Decision Variables**
  - Our optimization problem utilizes binary decision variables to represent family attendance on specific workshop days

- **Discrete Workshop Days**
  - Families can only attend the workshop on a specific and discrete day
  - No fractional attendance

- **Integer Constraints & Objective Functions**
  - Number of people and Associated Costs are Integer values

Problem Approach

Problem Formulation - Variables

Variables (Binary)

1. **Family Attendance Matrix $X_{i,j}$**

family_id	Day 1	Day 2	Day 3	Day 4	Day 5
0	$x_{1,1}$	$x_{1,2}$	$x_{1,3}$	$x_{1,4}$	$x_{1,5}$
1	$x_{2,1}$	$x_{2,2}$	$x_{2,3}$	$x_{2,4}$	$x_{2,5}$
2	$x_{3,1}$	$x_{3,2}$	$x_{3,3}$	$x_{3,4}$	$x_{3,5}$
3	$x_{4,1}$	$x_{4,2}$	$x_{4,3}$	$x_{4,4}$	$x_{4,5}$
4	$x_{5,1}$	$x_{5,2}$	$x_{5,3}$	$x_{5,4}$	$x_{5,5}$

Problem Approach

Problem Formulation - Constraints

Constraints

1 Family Attendance

- Each family is restricted to attending only one session to ensure equitable distribution of workshop slots among all families

2 Number of People per Day

- Sessions are optimized to accommodate a minimum of 125 to a maximum of 300 people per day

3 Total Cost

- For model operational efficiency, minimum and maximum bounds for total cost are set

Problem Approach

Compensation is offered to families that don't get their top preference choice

Choice	Gift Card	Santa's Buffet	Santa's Buffet Value (per family member)	Total Cost
Choice 0	$0	NA	NA	No consolation gifts
Choice 1	$50	NA	NA	$50
Choice 2	$50	25% off	$9	$50 + $9 * Number of Family Member
Choice 3	$100	25% off	$9	$100 + $9 * Number of Family Member
Choice 4	$200	25% off	$9	$200 + $9 * Number of Family Member
Choice 5	$200	50% off	$18	$200 + $18 * Number of Family Member
Choice 6	$300	50% off	$18	$300 + $18 * Number of Family Member
Choice 7	$300	Free	$36	$300 + $36 * Number of Family Member
Choice 8	$400	Free	$36	$400 + $36 * Number of Family Member
Choice 9	$699	Free	$36	$699 + $36 * Number of Family Member
Otherwise	$898	Free	$36	$898 + $36 * Number of Family Member

Problem Approach

Problem Formulation - Objective Functions

Objective Function

1 **Cost Functions**

- Our objective is to **minimize the total cost** incurred by assigning families to their preferred workshop days
  - For **each family**, calculate the **cost associated with each of their preference choices**
  - Calculate the cost for each family **based on the day they are assigned to**
  - Loop through all families, and **sum up the total costs** incurred across all families
  - **Minimize the total cost** to optimize the assignment of families to workshop days

Problem Approach

Constructing Cost Functions - If-Else Function

If-Else Function

- Constructed a **if-else function** to **directly calculate the cost within the function**
- Define a function **cost_per_family(i, j)** to calculate the cost c for assigning family i to workshop day j, incorporating the number of people associated with family i
- Determine the cost based on the preferred choices of family i, using **df[i, "choice_k"]** to represent the k-th preference choice

```
function cost_per_family(i,j)
    if j == df[i, "choice_0"]
        c = 0
    elseif j == df[i, "choice_1"]
        c = 50
    elseif j == df[i, "choice_2"]
        c = 50 + 9 * num_people(i)
    elseif j == df[i, "choice_3"]
        c = 100 + 9 * num_people(i)
    elseif j == df[i, "choice_4"]
        c = 200 + 9 * num_people(i)
    elseif j == df[i, "choice_5"]
        c = 200 + 18 * num_people(i)
    elseif j == df[i, "choice_6"]
        c = 300 + 18 * num_people(i)
    elseif j == df[i, "choice_7"]
        c = 300 + 36 * num_people(i)
    elseif j == df[i, "choice_8"]
        c = 400 + 36 * num_people(i)
    elseif j == df[i, "choice_9"]
        c = 699 + 36 * num_people(i)
    else
        c = 898 + 36 * num_people(i)
    end
    return c
end
```

Problem Approach

Deciding on the optimization solver - GLPK v.s. HiGHS

GLPK

- Well established solver that has **been around for decades**
- Larger user base
- **Reliable** and **stable** method for various types of optimization problem

HiGHS

- Newer solver designed for **performance**
- Efficient in solving **large/complex models**
- Utilizes **parallel computing resources** more effectively

Testing & Validation

Start with 10 families and 10 days

- Total Cost - $3,163
- 10 families, 39 people
- Cost may be comparatively high because not everyone have choices in the top 10 days

Day	1	2	3	4	5	6	7	8	9	10
No. People	8	2	4	6	4	0	7	0	0	8
Percentage	**20.51%**	**5.13%**	**10.26%**	**15.38%**	**10.26%**	**0.00%**	**17.95%**	**0.00%**	**0.00%**	**20.51%**

Scale to 100 families and 10 days

- Total Cost - $33,848
- 100 families, 397 people
- Families are more evenly distributed as we have more options now.

Day	1	2	3	4	5	6	7	8	9	10
No. People	50	45	50	49	50	12	25	44	22	50
Percentage	**12.59%**	**11.34%**	**12.59%**	**12.34%**	**12.59%**	**3.02%**	**6.30%**	**11.08%**	**5.54%**	**12.59%**

Results

Key findings and final cost results from our model

- **Optimal Solution:**
  - Total Cost: $43,622

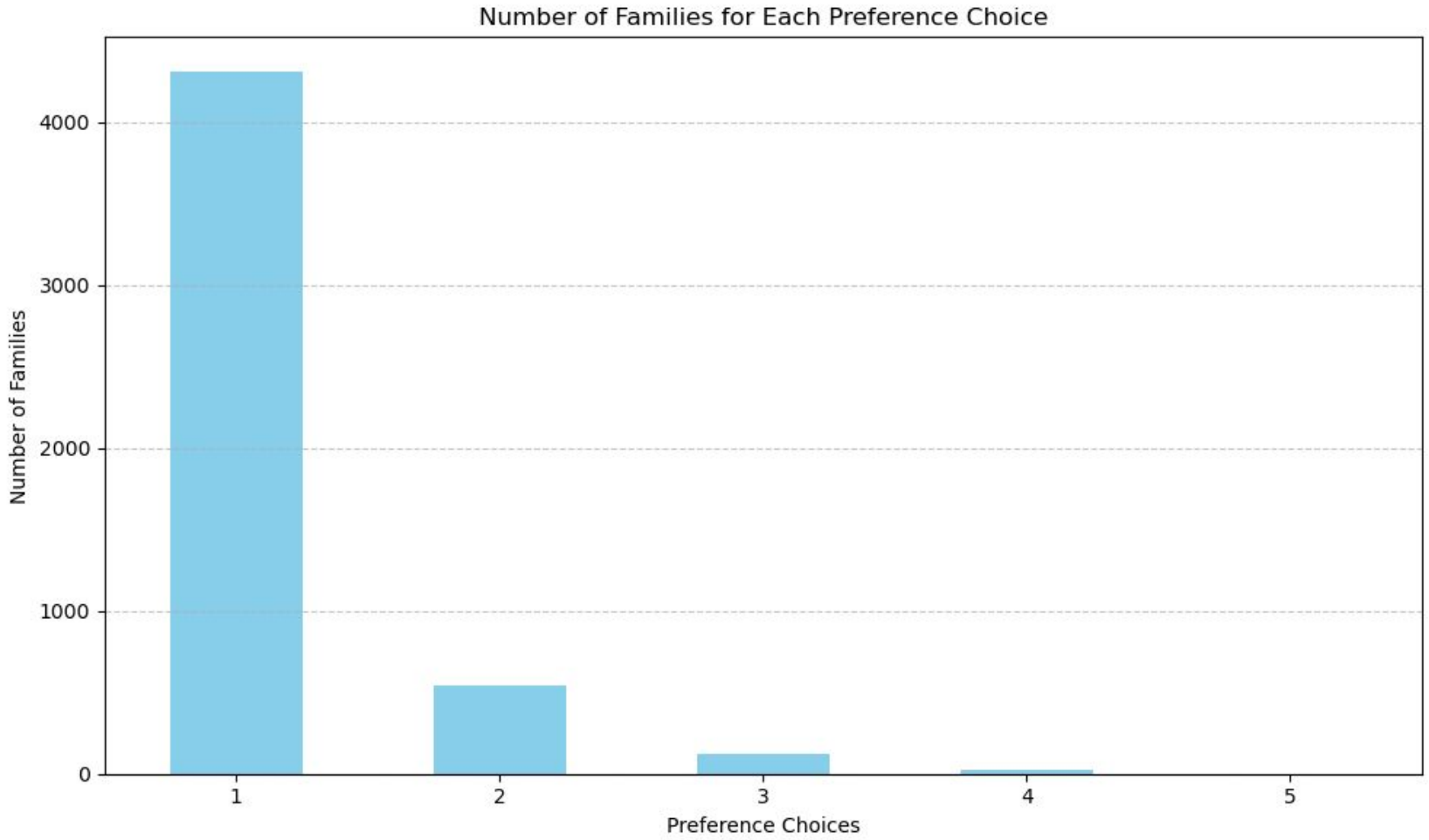
Tour Day	# of People	Tour Day	# of People
Day 62	125	Day 24	300
Day 63	125	Day 25	300
Day 64	125	Day 26	300

- **Adjusting the min/max number of people (75, 500):**
  - Total Cost: $15,225

Tour Day	# of People	Tour Day	# of People
Day 62	75	Day 24	355
Day 63	75	Day 25	415
Day 64	75	Day 26	354

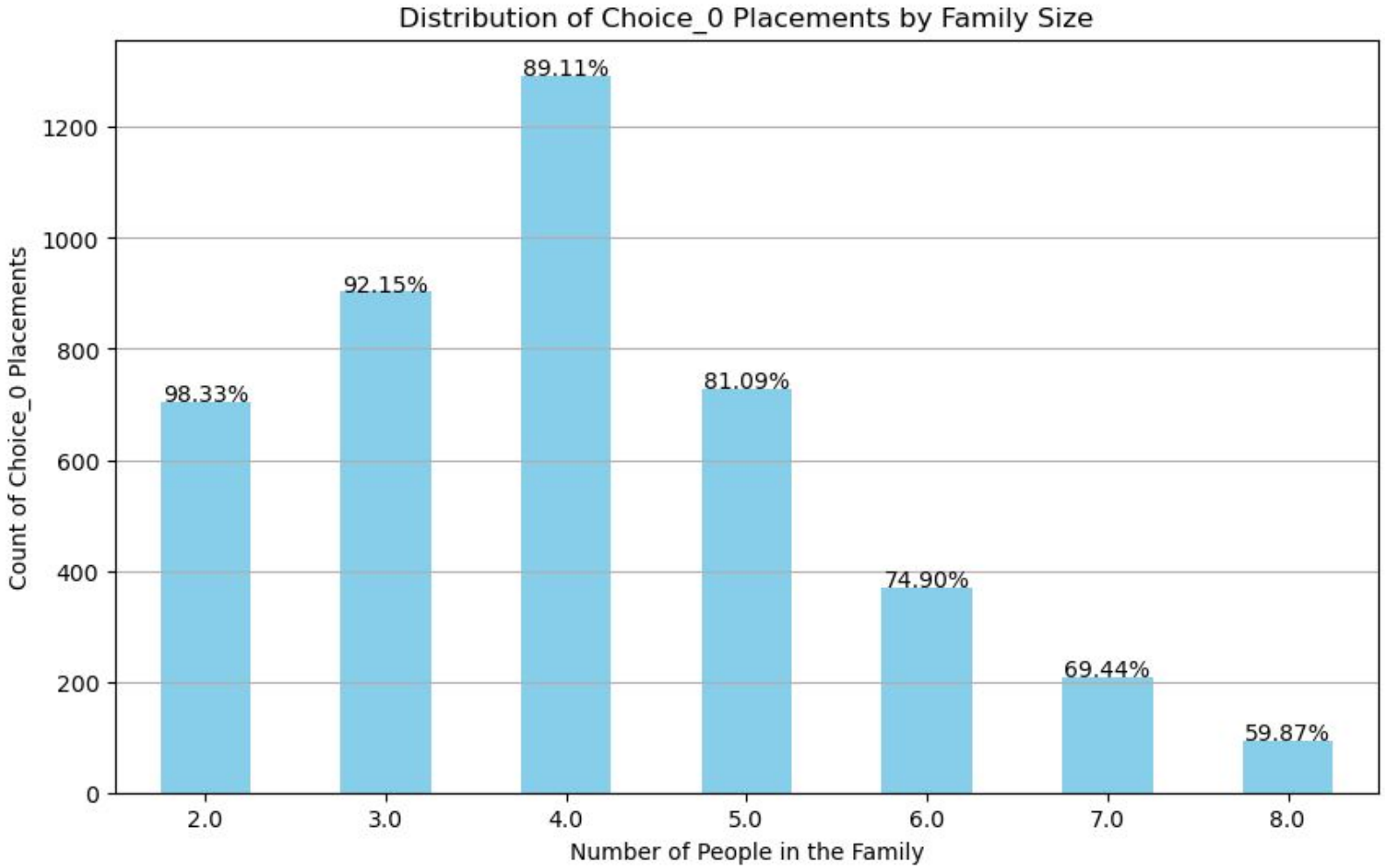
Results

Families predominately were able to get their first choice



Results

Families with less number of people are more likely to get placed in choice_0



Next Steps

Limitations, model efficiency improvements, and business value

Next Steps

A	Further Cost Considerations	1. Depending on the number of people attending a workshop on a given day, there can be a separate cost for cleaning fees, loss of revenue from Gift Shop due to too many shoppers.
B	Model Improvement	2. Tuning parameters based on results, pre-computing a cost matrix, using a warm start, and creating sparse representation of choices.
C	Business Value	3. Derive the amount that our model saves Santa compared to if he were to randomly assign the families to each day.