Service Business Workforce Optimization

Group 10: Alex Chin, Blenda Hoang, Billy Marin, Ruiyuan Luan, Shuang Peng (Penny)

Industry & Background

The beauty salon industry is a massive, growing industry within the United States. One major segment of the industry is the nail salon, where nail care services such as manicures and pedicures are provided. In 2020, the U.S. nail salon industry was estimated to be worth approximately $6.5 billion and comprised 27,963 different nail salons. Most nail salons in the U.S. are known as “Mom & Pop” establishments, meaning they are small, family or individually-owned businesses. According to a 2019 report, the average nail salon generates approximately $287,000 in gross revenues, with a 17% profit margin. However, it is important to note that one major challenge in studying the industry are the missing data points. Being a historically cash-based business, wages often go unreported and salons do not capture all workers on their payroll. Thus, it is believed that figures are often underestimated within the industry.

Another interesting component of the nail industry is its ties to Vietnamese culture, dating back to the Vietnam war. It is estimated that almost 40% of all nail technicians in the United States are of Vietnamese descent. In this case study, we will be trying to solve a problem proposed by Posh Nails, a small nail salon, owned and operated by a Vietnamese woman named Amanda.

Problem Statement

Posh Nails is a nail salon located in Indiana, operating 7 days/week. The salon provides several services varying from different manicure and pedicure experiences, artificial nails, dip nails, and special packages for kids. The work is performed by 5 experienced technicians and a pool of apprentice technicians. Experienced technicians earn more profit per shift because they can perform more expensive services for clients, such as dip nails, for example. The only time apprentice technicians are only utilized is when experienced technicians have the day off and customer demand cannot be met by the other experienced technicians.

With this information in mind, Amanda seeks to develop workforce plans each week. The primary goal is to balance the workload of the experienced technicians, ensuring each technician is assigned to the same number of shifts per week. The owner's secondary goal is to maximize profits, and to do so, she must reduce the number of times apprentice technicians are on the schedule.

Posh Nails' Goals
1. Balance the workload of the experienced technicians
2. Maximize profit based on staffing

Input Data

Chart #1

# of Shifts Required per Day

Mon	Tue	Wed	Thu	Fri	Sat	Sun
4	4	5	6	6	8	7

Chart #1 indicates the number of shifts required for each day in the week to meet customer demand.

Chart #2

Availability (Each Worker Has Two Days Off in a Week)

	Penny	Billy	Alex	Blenda	Ruiyuan
Mon			❌		
Tue					❌
Wed	❌			❌	
Thur			❌		
Fri	❌				
Sat		❌			❌
Sun		❌		❌	

Chart# 2 indicates each experienced nail technician's availability in which ❌ represents the nail technician is not available on that day.

Chart #3

Profits Earned from Nail Technicians and Apprentices of Each Shift	
Penny	$100
Billy	$90
Alex	$85
Blenda	$110
Ruiyuan	$90
Apprentice	$60

Chart #3 indicates the profits that the nail shop can earn from one shift per technician

Decision Variables

Availability of Experienced Technicians on each day:

Xij

- i = {1 … 5}
- j = {1 … 7}
- $\begin{cases} 1 \text{ if worker i scheduled to shift j} \\ 0 \text{ otherwise} \end{cases}$

Total # of Apprentice Technicians Working per Day:

- Yj, j = {1 … 7}

Total Profits Earned per Day:

- totprofits = (sum(profits)*totshifts)

Total shifts of each experienced technician:

- totShifts

minShift and maxShift:

- minShift = m.addVar(name='minShift')
- maxShift = m.addVar(name='maxShift')

Constraints

Shift requirements, # of shifts required of each day:

- Shift Requirement = ∑Xij +∑Yj

Total shifts of each workers:

- totShifts[w] == technician.sum(w,'*') for w in workers

Total profits of the nail shop:

- totprofits == (sum(profits[i]*totShifts[i] for i in workers) +

  (novice.sum())*noviceprofit

Minshift and Maxshift constraints:

- min_constr = m.addGenConstrMin(minShift, totShifts, name='minShift')
- max_constr = m.addGenConstrMax(maxShift, totShifts, name='maxShift')

Objective Function

Minimize the workload imbalance of experienced technician

- minimize : maxShift - minShift

Maximize weekly profits by ensuring appropriate amount of staff is scheduled

- minimize : -totprofits

Gurobi Model Deployment

Import libraries:

```
#This command imports the Gurobi functions and classes.
import gurobipy as gp
from gurobipy import GRB

import pandas as pd
from pylab import *
import matplotlib
import matplotlib.pyplot as plt
```

Input data:

```
# toal number of shifts required for each day
day, shiftRequirements = gp.multidict({
  "Mon": 4,
  "Tue": 4,
  "Wed": 5,
  "Thu": 6,
  "Fri": 6,
  "Sat": 8,
  "Sun": 7})
```

```
# Worker availability: defines on which day each nail technician worker is available.
availability = gp.tuplelist([
('Penny', 'Mon'), ('Penny', 'Tue'), ('Penny', 'Thur'), ('Penny', 'Sat'),('Penny', 'Sun'),
('Billy', 'Mon'), ('Billy', 'Tue'), ('Billy', 'Wed'), ('Billy', 'Thur'),
('Billy', 'Friday'), ('Alex', 'Tue'), ('Alex', 'Wed'), ('Alex', 'Friday'),
('Alex', 'Sat'), ('Alex', 'Sun'), ('Blenda', 'Mon'), ('Blenda', 'Tue'),
('Blenda', 'Thur'), ('Blenda', 'Fri'), ('Blenda', 'Sat'), ('Ruiyuan', 'Mon'),
('Ruiyuan', 'Wed'), ('Ruiyuan', 'Thur'), ('Ruiyuan', 'Fri'), ('Ruiyuan', 'Sun'),
])
```

```
workers, profits = gp.multidict({
  'Penny': 100,
  'Billy': 90,
  'Alex':85,
  'Blenda':110,
  'Ruiyuan':90,
})

noviceprofit = 30
```

Set model and decision variables:

```
# Create initial model.
m = gp.Model("nail")
```

```
#variable - total shifts each technician assigned to a day
technician = m.addVars(availability, vtype=GRB.BINARY, name="technician")
```

```
#variable - total shifts of novice assigned to each day
novice = m.addVars(day, vtype=GRB.INTEGER, name="novice")
```

```
#total shifts each technician assigned to next week
totShifts = m.addVars(workers) #, name="TotShifts"
```

```
#totnovice = m.addVar(name='totnovice')
```

```
#total profits the nail shop earned next week
totprofits = m.addVar(name='totprofits')
```

Add constraints:

```
con1 = m.addConstrs((technician.sum('*',s) + novice[s] == shiftRequirements[s] for s in day), name='shiftRequirement
```

```
con2 = m.addConstrs((totShifts[w] == technician.sum(w,'*') for w in workers), name='totShifts')
```

```
con3 = m.addConstr(totprofits == (sum(profits[i]*totShifts[i] for i in workers) + (novice.sum())*noviceprofit), \
                   name='profits')
```

```
<ipython-input-15-964ceeff06a6>:1: DeprecationWarning: Calling np.sum(generator) is deprecated, and in the future w
ill give a different result. Use np.sum(np.fromiter(generator)) or the python sum builtin instead.
  con3 = m.addConstr(totprofits == (sum(profits[i]*totShifts[i] for i in workers) + (novice.sum())*noviceprofit), \
```

```
minShift = m.addVar(name='minShift')
maxShift = m.addVar(name='maxShift')

min_constr = m.addGenConstrMin(minShift, totShifts, name='minShift')
max_constr = m.addGenConstrMax(maxShift, totShifts, name='maxShift')
```

Optimization result:

```
m.ModelSense = GRB.MINIMIZE
```

```
m.setObjectiveN(maxShift - minShift, index=1, priority=1, name='Fairness')
```

```
m.setObjectiveN((-totprofits), index=1, priority=2, name='profit')
```

```
m.write('nail.lp')
```

```
m.optimize()
```

Business Insight & Recommendations

After running the Gurobi model, our optimal solution can be displayed by plotting the technician's weekly work schedule. As shown below, the experienced technicians' workload is balanced, with each working 5 shifts per week and fulfilling Amanda's first objective. Apprenticed technicians are then called on for support, for days in which demand exceeds experienced technician supply, such as Thursday. Additionally, with this scheduling mix, profits are maximized at $3,005 per week, which fulfills the owner's second objective. Lastly, if Amanda is looking to hire, we recommend her to target candidates who can work the latter half of the week, as that is when demand is greatest and more apprenticed technicians are being called in for support. This will help ensure customer demand is consistently met and can help boost profits each week.

Weekly Schedule

	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday	Shifts Worked
Penny	X	X	O	X	O	X	X	5
Ruiyuan	X	O	X	X	X	O	X	5
Billy	X	X	X	X	X	O	O	5
Blenda	X	X	O	X	X	X	O	5
Alex	O	X	X	O	X	X	X	5
Apprentice	0	0	2	6	4	5	4	

Total Profits $ 3,005