BANA4095: Decision Models - Spring 2021
Clifton Running Shoes Case

Dr. Charles R. Sox
Associate Dean - Impact & Partnerships
Professor of Operations & Business Analytics

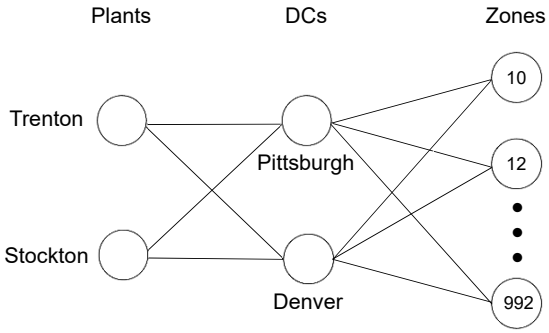
1

Outline

- CRS Case
  - » Discussion
  - » General Concepts
  - » Model Formulation
- Reading Excel Files
  - » `openpyxl` Package

2

CRS Supply Chain



3

Model Discussion

- Plants
- Customer Zones
- Candidate DCs
- Costs
  - » Inbound, Outbound, Handling
- Customer Service
  - » Two-day delivery
- Decision Variables?
- Objective?
- Constraints?

4

Model Formulation

Notation:

y_j = 1 if DC j is selected/open; 0 if not
x_{ij} = number of units shipped from node i to node j
c_{ij} = cost per unit of shipping from node i to node j
s_i = supply capacity at plant i
d_j = demand at customer zone j

5

Model Formulation

$$\text{Min} \sum_{\text{all arcs}} c_{ij}x_{ij} + \sum_{\text{all arcs}} c_{jk}x_{jk}$$

s.t.

$$\sum_{\text{arcs in}} x_{ij} - Myj \le 0 \qquad \text{for each DC } j$$

$$\sum_{\text{arcs out}} x_{ij} \le s_i \qquad \text{for each plant } i$$

$$\sum_{\text{arcs in}} x_{jk} = d_k \qquad \text{for each customer zone } k$$

$$\sum_{\text{arcs in}} x_{ij} - \sum_{\text{arcs out}} x_{jk} = 0 \qquad \text{for each DC } j$$

y_j in {0,1} for all j
x_{ij}, $x_{jk} \ge 0$ for all i, j, k

6

Customer Service

- Two-day delivery
- Transit times table
- How to compute percentage of shipments delivered within two days?

7

Reading Data from an Excel File

- `openpyxl` Package
  - » Provides functionality for reading and writing Excel files in Python
  - » Included in the Anaconda distribution
  - » See `openpyxl.ipynb` for more details . . .

```
import openpyxl as opx
wb = opx.load_workbook('CRS_data.xlsx')
ws = wb['Forecasted Demand']
demand = dict()
for row in ws.values:
    (zone,dmd) = row
    demand[zone] = dmd
del demand['Customer Zone']
```

8

Summary

- CRS Case
  - » Discussion
  - » General Concepts
  - » Model Formulation
- Reading Excel Files
  - » `openpyxl` Package

9