40.016: The Analytics Edge
Week 12 Lecture 1 (Optional)

PRESCRIPTIVE ANALYTICS WITH JULIA (PART 1)

Term 5, 2022

SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN

Outline

Outline

Three types of analytics

- Descriptive Analytics
- Predictive Analytics
- Prescriptive Analytics

Descriptive Analytics

Using data aggregation and data mining to provide insights into the past and answer:

"What has happened?"

Example:

- Suppose an unusually high number of people are admitted to the emergency room in a short period of time.
- Descriptive analytics tells you that this is happening and provides real-time data with all the corresponding statistics, such as date of occurrence, volume, patient details.

Predictive Analytics

Using statistical models and forecasting techniques to understand the future and answer:

"What could happen?"

Example:

- Back to our hospital example: predictive analytics may forecast a surge in patients admitted to the ER in the next several weeks. Based on patterns in the data, the illness is spreading at a rapid rate.

Prescriptive Analytics

Using optimization and simulation algorithms to advise on possible outcomes and answer:

"What should we do?"

Example:

- Back to our hospital example: now that you know the illness is spreading, the prescriptive analytics tool may suggest that you increase the number of staff on hand to adequately treat the influx of patients.

Outline

Key features of Julia

- General-purpose programming language
- Dynamic-type system
- Open source
- Performance approaching that of statically-typed languages (e.g., C/C++)
- Interfaces with other languages, such as R and Python
    - `JuliaCall`: Provides an R interface to Julia
- Supports parallel and distributed computing
- Built-in package manager

History of Julia

- Work on Julia started in 2009
- Julia Computing founded in 2015
- In 2017, Julia was used for petascale computing (i.e., 10^{15} floating point operations per second)
- Julia 1.0 released in 2018

Editors of Julia

- Juno
- VSCode
- Jupyter Notebooks
  - The easiest way to install Jupyter and Python simultaneously is to use Anaconda
  - Requires IJulia (package that provides the connection between Jupyter and Julia)

Installation

Step 1. Install Julia (`https://julialang.org/`)

Step 2. Run the Julia application (double-click on it); a window with a `julia>` prompt will appear.

Step 3. Install IJulia: at the prompt, type:

```
using Pkg
Pkg.add("IJulia")
```

Step 4. Running the IJulia Notebook: at the prompt, type:

```
using IJulia
notebook()
```

to launch the IJulia notebook in your browser.
(Note that the first time you run `notebook()`, it will prompt you for whether it should install Jupyter.)

IJulia and Jupyter

How to launch Jupyter for Julia?

Option 1: Type `jupyter notebook` in the terminal

Option 2: Type the following commands in Julia (`julia>` prompt)

```
using IJulia
notebook()
```

JuMP

JuMP is a domain-specific modeling language for mathematical optimization embedded in Julia.

Key features:

- Modeling language for mathematical optimization embedded in Julia
- Supports many solvers (both open-source and commercial)
- See the Installation Guide for a list of solvers
- Several classes of problems
  - LP: Linear Programming
  - MILP: Mixed-Integer Linear Programming
  - Nonlinear Programming ...

Workflow for creating a model

- Load JuMP and a solver / optimizer (e.g., GLPK)

```
import Pkg
Pkg.add("JuMP")
Pkg.add("GLPK")
```

  see file `Packages.ipynb`.

- Create a model with the `Model()` function
- Add variables (`@variable()`)
- Add objective (`@objective()`)
- Add constraints, if any (`@constraint()`)

Workflow for solving a problem

- Use the `optimize!()` function
- Check the status of the solution (`termination_status()`)
- Check the obtained objective function (`objective_value()`)
- Check the obtained solution (`value.()`)

Example 1

$$
\begin{aligned}
\min_{x,y} \quad & x + y \\
\text{s.t.} \quad & x + y \leq 1 \\
& x, y \geq 0 \\
& x, y \in \mathbb{R}
\end{aligned}
$$

File `Example 1.ipynb`.

Example 2

$$\begin{aligned} \min_x \quad & c^T x \\ \text{s.t.} \quad & Ax = b \\ & x \geq 0 \\ & x \in \mathbb{R}^n \end{aligned}$$

File `Example 2.ipynb`.

Example 3

$$
\begin{aligned}
\min_{x,y} \quad & c^T x + d^T y \\
\text{s.t.} \quad & Ax + By = f \\
& x, y \geq 0 \\
& x \in \mathbb{R}^n, y \in \mathbb{Z}^p
\end{aligned}
$$

File `Example 3.ipynb`.

Outline

Attributes of revenue management systems

- They find ways to increase revenue without necessarily changing products;
- Prices are better aligned to what customers are willing to pay;
- Computational power allows to solve large-scale optimization models.

Aviation industry

- In the late 1970s, deregulation in the airlines industries provided opportunities (and challenges) for many of the airline carriers to increase profit and stay ahead of the competition.
- Managing the seats sold effectively was realized to be important, so revenue management techniques were used to increase revenue.
- Today, revenue management and the use of analytics for this purpose is affecting many industries, including hotels and online retailers.
- Key challenge:

sell the right product to the right customer at the right time for the very right price

Key features of the aviation industry

- Fixed capacity (i.e., number of seats)
- High fixed costs but low variable costs
- Variety of customer types (e.g., price sensitive, luxury travellers)
- Ability to sell tickets to customers without seeing what other customers paid

– made it suitable for revenue management

Revenue management in aviation industry

Note: The scale of this problem can be huge, since

- reservations are made often months in advance to a day before
- there are multiple departures for an airline carrier ...

Challenges:

- Traffic management
- Discount allocation
- Pricing
- Overbooking

Outline

Discount allocation

- The discount allocation is the problem of determining the number of discount fares to offer for the flights.
- The tradeoff is between offering discounts so as to fill all seats or not offering discounts, but with the risk of having some empty seats (which can be taken by late-arriving high-revenue passengers).

Discount allocation (cont'd)

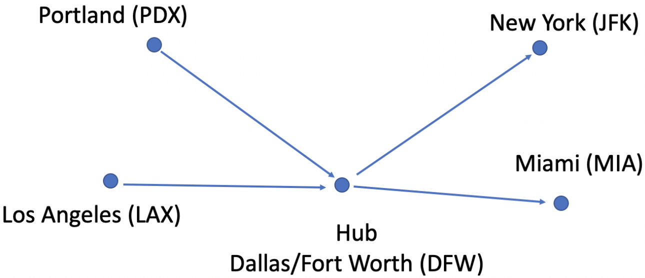


Figure: Four spoke network centred in Dallas/Fort Worth International Airport.

- There are four types of passenger routes in this network.
- Furthermore, there are two classes of passengers, one who books in advance and wants to pay less and a second class who books late but is willing to pay a lot more.

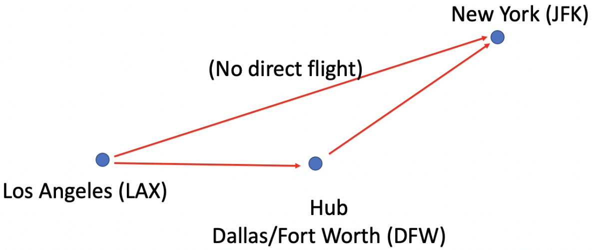
Discount allocation (cont'd)

This inspired the airlines offer two products:

- More expensive tickets that can be purchased anytime, with no restrictions, and fully refundable
- Cheaper tickets, to be bought at least 3 weeks in advance, with penalties for changes

Problem formulation: Assumptions

- Prices have already been decided by the marketing department
- We are interested in deciding how many seats to reserve in each fare class and itinerary
- For simplicity, we consider three flights: LAX $\rightarrow$ DFW, LAX $\rightarrow$ JFK, and DFW $\rightarrow$ JFK



Problem formulation: Data

Table: Flight capacity of LAX $\rightarrow$ DFW and DFW $\rightarrow$ JFK flights.

Flight	Capacity
LAX $\rightarrow$ DFW	300
DFW $\rightarrow$ JFK	200

Problem formulation: Data (cont'd)

Table: Revenue and demand for each fare of LAX $\rightarrow$ DFW, LAX $\rightarrow$ JFK, and DFW $\rightarrow$ JFK flights. The last column reports the decision variables for each leg.

Flight	Fare	Revenue ($)	Demand	Decision variable
LAX $\rightarrow$ DFW	Regular	100	20	x_1
	Discount	90	40	x_2
	Saver	80	60	x_3
LAX $\rightarrow$ JFK	Regular	215	80	y_1
	Discount	185	60	y_2
	Saver	145	70	y_3
DFW $\rightarrow$ JFK	Regular	140	20	z_1
	Discount	120	20	z_2
	Saver	100	30	z_3

Maximizing revenue through LP

$$
\begin{aligned}
\max \quad & 100x_1 + 90x_2 + 80x_3 + 215y_1 + 185y_2 + 145y_3 + 140z_1 + 120z_2 + 100z_3 \\
\text{s.t.} \quad & x_1 + x_2 + x_3 + y_1 + y_2 + y_3 \leq 300 \\
& y_1 + y_2 + y_3 + z_1 + z_2 + z_3 \leq 200 \\
& 0 \leq x_1 \leq 20 \\
& 0 \leq x_2 \leq 40 \\
& 0 \leq x_3 \leq 60 \\
& 0 \leq y_1 \leq 80 \\
& 0 \leq y_2 \leq 60 \\
& 0 \leq y_3 \leq 70 \\
& 0 \leq z_1 \leq 20 \\
& 0 \leq z_2 \leq 20 \\
& 0 \leq z_3 \leq 30
\end{aligned}
$$

File `seats.ipynb`.

Optimal solution

The optimal solution is: $x_1^* = 20$, $x_2^* = 40$, $x_3^* = 60$, $y_1^* = 80$, $y_2^* = 60$, $y_3^* = 40$, $z_1^* = 20$, $z_2^* = 0$, $z_3^* = 0$, with a total revenue of USD 47,300.

A greedy method would start by allocating seats to the customers yielding the highest revenue, and then focus on the remaining customers. Such method would yield a revenue of USD 45,800.

What would a greedy method do?

- Give priority to the highest revenue customers and greedily allocate them.
- In the problem described above, we could set $y_1 = 80$, $y_2 = 60$, and $y_3 = 60$ (since the capacity of the DFW $\rightarrow$ JFK is only 200).
- We would still have 100 seats on the LAX $\rightarrow$ DFW flight, so we could set $x_1 = 20$, $x_2 = 40$, and $x_3 = 40$.
- The total revenue in this case would be $45,800, which is about 3% less than the optimal revenue.
- Aggregating such loss over all flights and all fare classes would lead to a significant amount of money.

Practical implementation

- As soon as regular priced ticket are sold, only discount tickets are available though it is possible that some customers might still be willing to buy at a regular price.

  – Airlines use “nesting control”, where seats reserved for discount classes are made available to more expensive classes.

- The advantage of an LP model is that it provides a shadow price for each constraints (dual variable). This reflects the revenue value of the last seat on each leg of the flight for the capacity constraints.

- The data are often changing with time, so dynamic models with randomness in demand must be incorporated in such implementation.

Outline

Prescribing the right price

- One of the key aspects of revenue management is pricing.
- Consider the following demand curve for a single product:

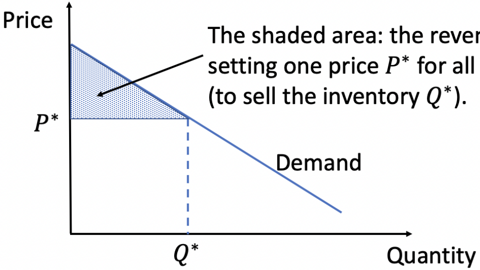
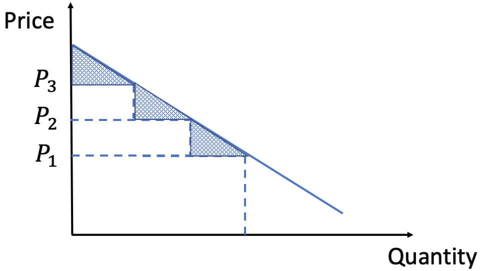


Figure: Q^*: the amount in inventory; P^*: the maximum price that would allow all inventory to be sold.

Prescribing the right price (cont'd)

- With a revenue management approach, different prices are set for the same product. – The product looks the same to all of the customers (e.g., airline seats, standard hotel rooms), but the goal is to get the customers who are willing to pay more to do so.
- Suppose we can set three prices for the same product. Then, we can reduce the amount of revenue lost.



Pricing products with a multinomial Logit model

Consider a set of products $\{1, \ldots, n\}$ and an outside option denoted by 0. The goal of the decision-maker is to price products $1, \ldots, n$ through the following optimization problem:

$$\max_{p_1 \geq 0, \ldots, p_n \geq 0} \sum_{i=1}^{n} (\underbrace{p_i}_{\text{price}} - \underbrace{c_i}_{\text{cost}}) \underbrace{\mathbb{P}(\text{choosing product } i \text{ given prices } \bar{P})}_{\text{Obtained from logit model}},$$

where

$$\mathbb{P}(\text{choosing product } i \text{ given prices } \bar{P}) = \frac{e^{v_i - p_i}}{\sum_{j=0}^{n} e^{v_j - p_j}},$$

where p_i is the price of the i-th product and v_i its corresponding evaluation.

Pricing products with a multinomial Logit model (cont'd)

The optimization problem can be rewritten as:

$$\max_{p_1 \geq 0, \ldots, p_n \geq 0} \sum_{i=1}^{n} \frac{(p_i - c_i) e^{v_i - p_i}}{\sum_{j=0}^{n} e^{v_j - p_j}},$$

where $v_0 = p_0 = 0$ are the evaluation and price of the default (outside) option.

- This is not an easy problem to solve.
- The problem can be re-formulated and solved using choice probability variables (see the derivation in the teaching notes).
- The implementation is provided in the file `pricing.ipynb`.

References: Julia (with links)

- Julia https://julialang.org/
- IJ https://github.com/JuliaLang/IJulia.jl
- IJulia https://julialang.github.io/IJulia.jl/stable/manual/installation/
- JuMP https://jump.dev/JuMP.jl/stable/

References: Revenue management

- Teaching notes.
- Bertsimas, D., Allison, K. O., & Pulleyblank, W. R. (2016). *The Analytics Edge.* Dynamic Ideas LLC.
- Smith, B. C., Leimkuhler, J. F., & Darrow, R. M. (1992). *Yield management at American airlines.* Interfaces, 22(1), 8-31.