Business Decision Analytics under Uncertainty
Spring 2017, Professor Eckstein
Homework 7

Due Wednesday, March 29

Q1: Managing a spare parts supply

Your factory uses a supply of parts at a steady rate of 10 per month. You have inventory capacity to hold 40 of the parts, and holding parts in inventory for one month incurs a direct cost of $0.50 per part per month. Your firm also has an internal cost of funds of 0.5% per month. At the beginning of each month, you may place an order for parts, which cost $200 each, plus an overhead cost of $65 for each order placed.

You have very stringent quality requirements for the parts, which your supplier, despite being the best one available, often does not meet. Each part ordered has an independent 6% chance of not meeting your standards. You immediately return such rejected parts for a full refund, but shipping them back to the supplier costs $7 per part.

Along with parts already in inventory, parts ordered at the beginning of the month may be used to meet the current month's usage, so long as they are not rejected. If you fall short of the usage requirement for a given month, each part you are "short" that month incurs a $450 opportunity cost. With a Python program, use dynamic programming to compute the part ordering policy that minimizes the expected present value of your costs over the next 12 months, under the constraint that you are not permitted to overflow your inventory capacity.

Assume that parts left in inventory at the end of the 12 months have a salvage value of $180 each, and you start with an inventory of 15 parts.

Hand in a hardcopy printout of your Python program and a printout of its output. Also submit your Python code to BlackBoard as an attachment under the "Assignments" tab and "Homework Assignment 7, Q1 Code".

Q2: Hotel yield management

You operate a small hotel with 20 rooms. For a particular day 10 days from now, call it day *D*, you do not have any rooms booked. You take room bookings exclusively through the website "Tripspedia.com". At the beginning of each day (including the morning of day *D*), you inform Tripspedia.com to post a price of either $100, $150, $200, $250, or $300 for rooms available on day *D*. You have performed a market analysis of past demand patterns and you estimate that, on day *t*, if you post a price of *P* dollars for rooms on day *D*, the number of rooms booked for day *D* is a Poisson random variable with mean

$$M(t,P) = \big(5 - 0.25(D - t)\big)(0.7)^{(P/50 - 2)}$$

For example, if there are two days to go in until day *D* (and hence $D - t = 2$), and you post a price of \$150, the number of rooms bookings for day *D* that customers will try to place on day *t* will be a Poisson random variable with mean

$$(5 - 0.25 \cdot 2)(0.7)^{(150/50 - 2)} = (4.5)(0.7)^{1} = (4.5)(0.7) = 3.15.$$

Assume that bookings are nonrefundable, there are no cancellations, and you never overbook the hotel, that is, each week, you only accept as many bookings as you have rooms left unreserved.

Write a Python program that uses dynamic programming to determine an optimal pricing policy for the 10 days up to (and including) day *D*. **Hint:** the stages should be the 10 days up to and including day *D* and the state should be the number of rooms that are not yet booked. The decision in each stage-state combination is the price you announce to Tripspedia.

Hand in a hardcopy printout of your Python program and a printout of its output. Also submit your Python code to BlackBoard as an attachment under the "Assignments" tab and "Homework Assignment 7, Q2 Code".

Note: this problem is a simplification of the actual pricing process used by airlines, hotels, and car rental companies. The actual process is more complicated because cancellations are possible and there may be multiple semi-interchangeable products, such as cars or rooms of varying sizes, or fares with varying restrictions or ancillary services (like free baggage versus charged baggage).