

Business Decision Analytics under Uncertainty
Spring 2018, Professor Eckstein
Homework 4
Due Wednesday, March 7

Solve each problem by hand using dynamic programming. Before solving each problem, answer the following:

- What are the *stages* in this problem?
- What defines the *states* within each stage?
- For each stage-state combination, what are the *decisions* you are choosing between?
- What is the *interpretation of the value function* $f_i(i)$ for this particular problem?
- Optionally, you may state the recursion that defines the value function f_i in terms of f_{i+1} (except in the last stage).

(For examples of the interpretation of the value function and the recursion, see the instructions for homework 3.) Next, solve the problem by hand using dynamic programming, showing your work. State the optimal solution and its value.

Q1: An Integer Knapsack problem

Suppose you have a knapsack that can hold 7 pounds, and items of the following type:

Type	Weight	Value
1	1	\$1
2	2	\$3
3	3	\$4

You would like to maximize the value of items placed in the knapsack, without exceeding its capacity of 7 pounds. You can place multiple copies of the same kind of item in the knapsack (for example, you fill the knapsack with 4 items of type 1 and 1 item of type 3).

By “following the stars” (*’s) in your solution, you should be able to identify *two* different optimal solutions. Please state both of them.

Q2: Airline Flight Planning

Problem 4 on page 270 of the text (with a table of data on page 271), solving only the simpler case with 4 flights (you do not have solve to the first, more complicated version with 6 flights). Note that the top heading in the table is mislabeled: it should be “profit”, not “profit per flight”. For each destination city, assume that it also possible to send zero flights there, in which case you have zero profit from that city. Make New York stage 3, Los Angeles stage 2, and Miami stage 1 (the correspondence of cities to stages can be shuffled arbitrarily, but it will be easier to grade the assignment if everybody uses the same ordering).