Session 2

Analysis and Synthesis

Array Arithmetic and Matrix Products

Last revised: July 6, 2011

Review of last time: Introduction

- Integration/Accumulation
  - "Running Sum"
  - Initial Value
- Differentiation/Differencing
  - "Running differences"
  - Inverse of accumulation
- Named parameters
  - Store constants in cells — "parameterize" your model
  - Make all uses of those parameters refer to those cells
  - Give those cells names — refer to them by name
  - Collect parameters into parameter blocks
  - Name the blocks
  - Names can have global or local scope
- References: relative and absolute

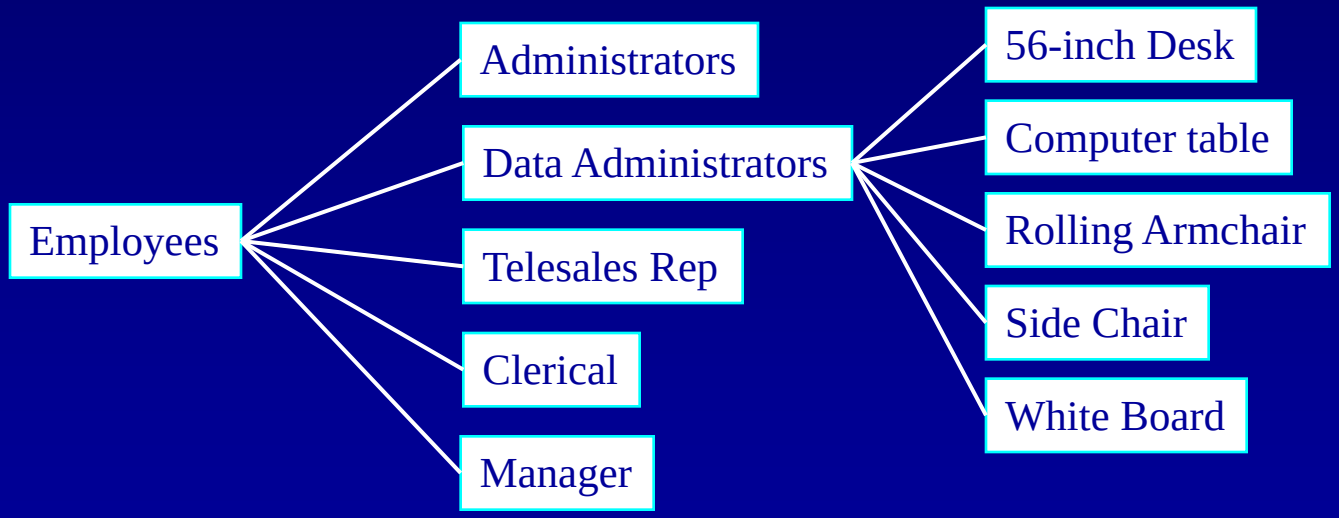
Analysis and synthesis

- Analysis
  - Separation of a whole into its component parts
  - Used for decomposing a modeled quantity into pieces
- Synthesis
  - Composition or combination of parts or elements to form a whole
  - Used for combining different contributing effects
- Why do it?
  - Sometimes it's easier to look at a problem by breaking it into parts and recombining the results
  - If a change is required, we can often change just one component, rather than the whole thing
  - As the model evolves, we often add or delete components without worrying about the others
  - This is the same advantage as that of object-oriented programming

Example of analysis and synthesis

The furniture requirements of a telesales department

- First analyze the categories of employees
- Next analyze their furniture requirements
- To find total requirements, add the components



Implementation

- When we decompose (analyze) we ask:
  - What are the components of the whole system with respect to a particular way of breaking it up?
  - Example: Divide work force into categories, and ask how much of each kind of furniture is needed in each category
- When we recombine (synthesize) we ask:
  - What is the total composition of the system, given the parts?
  - Example: If we know how much furniture is required for each category of employee, and we know how many employees we have in each category, we can roll up the total furniture spectrum
- Analysis and Synthesis are best implemented using Excel arrays

Arrays in Excel

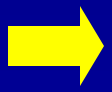
- An array is a rectangular range of cells
- Analysis and Synthesis best done with array manipulation
- It's possible to do this cell-by-cell, but that approach is:
  - Cumbersome
  - Error-prone
  - Hard to follow and maintain
- Better: Use arrays
  - An array formula returns an array that is spread over the cells in which you enter it
  - One array=>One formula
  - Faster computations
  - Sometimes you have to transpose the arrays to get the result you need
- How to enter an array formula
  - Windows: Control+Shift+Enter
  - Mac: Command+Enter or Control+Shift+Enter

Array arithmetic

- Arrays in Excel obey the usual laws of arithmetic for +, -, *, /, ^, …
- If the arrays are the same shape and size, perform operations cell-by-cell
- If the arrays are the same height (width) but one has only one column (one row), expand that array to match the other array

Matrix multiplication

- A *matrix* is represented by a rectangular range
  - An "m by n" matrix has m rows and n columns
  - In a matrix product A*B, the first factor A must have as many columns as the second factor B has rows
  - The product has the same number of rows as the first factor and the same number of columns as the second factor
- To multiply a 1 x p matrix by a p x 1 matrix:
  - Multiply each of the respective elements together and sum
- For different sizes, just break them up into 1 x n and n x 1



Readings: Matrices and Matrix Multiplication

Matrix multiplication examples

$$\begin{bmatrix} 1 & 3 & 2 \end{bmatrix} \begin{bmatrix} 2 \\ 3 \\ 1 \end{bmatrix} = \begin{bmatrix} 13 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \begin{bmatrix} 2 \\ 3 \end{bmatrix} = \begin{bmatrix} 8 \\ 18 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \begin{bmatrix} 2 & 1 \\ 4 & 3 \end{bmatrix} = \begin{bmatrix} 10 & 7 \\ 22 & 15 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 1 & 3 \end{bmatrix} \begin{bmatrix} 2 \\ 3 \end{bmatrix} = \begin{bmatrix} 8 \\ 18 \\ 11 \end{bmatrix}$$

Transposing matrices

- To transpose, just interchange rows and columns

$$\begin{bmatrix} 1 & 3 & 2 \end{bmatrix}^t = \begin{bmatrix} 1 \\ 3 \\ 2 \end{bmatrix}$$

$$\begin{bmatrix} 9 & 3 \\ 19 & 5 \\ 11 & 1 \\ 19 & 14 \end{bmatrix}^t = \begin{bmatrix} 9 & 19 & 11 & 19 \\ 3 & 5 & 1 & 14 \end{bmatrix}$$

- There are some simple rules for matrix products.
  If A and B are matrices and x and y are just numbers:
  - $(A^t)^t = A$
  - $(AB)^t = B^tA^t$
  - $(xA \pm yB)^t = xA^t \pm yB^t$

Application examples

- Your company wants to replace all its office furniture. You have the following data:
  - Who needs what pieces, by employee category
  - Price of each type of piece
  - How many employees are in each category

  How much will it cost to furnish the entire office?
- Under these same conditions, you're given a set of hiring streams for each category of employee
  - Assume that the information above applies
  - Determine how much you'll spend each month if hiring proceeds as planned

Reference reading

- Course readings
  - Array arithmetic (Readings: Array Arithmetic)
  - Dimensional Analysis (Readings: Dimensional Analysis)
  - Matrix products (Readings: Matrices and Matrix Multiplication)
  - Worksheet Functions (Readings: Worksheet Functions)
  - Names (Readings: Names)
  - Workbooks (Readings: Workbooks)
- On-line Help topics
  - Worksheet function `MMULT`
  - Worksheet function `TRANSPOSE`
  - Worksheet function `IF`
  - Worksheet function `SUM`
  - Entering an array formula

Preview of next time: Cushioning, Crowding and Quantization

- Modeling is inexact, so we use safety margins
- Sometimes you plan shortages
- Sometimes resources are available only in certain minimum size lots
- Excel functions that help with these issues:
  `CEILING`, `FLOOR`, `INT`, `ROUND`, `TRUNC`
- Circular references
  - Arise when loops of cells depend on each other
  - Break the circularity by solving the equations if possible