Data Analysis using Spreadsheets 2

- Finding Optimum Solutions
- Optimisation Examples
- Matrix Operations
- Financial Functions
- Cross Tables and Look-up Functions
- Scenarios
- Testing and Validating Spreadsheets

* not covered in semester 1

Data is cool

- You can use data to
  - Discover hidden patterns
  - Learn about the world
  - Do prediction
  - Quality control

Upward trend of the amount of CO$_2$ in the atmosphere

Data can help to make complicated decisions

- Research project: To understand how micro-climate affects forest regeneration.

- Sensors: Wind direction, wind speed, temperature, humidity etc.
- Insufficient solar power, need to make a decision on which sensor combinations (there are many) to use

Decision problems

- Quite often we need to take constraints into account when making decisions
- Constraints can arise because of limitation in
  - Energy
  - Time
  - Budget
- A systematic method to make decisions is to formulate an optimisation problem

Optimisation problem

Becky’s dream is to study Engineering at UNSW (obviously)

Maximise ATAR score, spending...

- $x_1$ hours/week on English
- $x_2$ hours/week on Maths
- $x_3$ hours/week on Physics
- $x_4$ hours/week on Hockey
- $x_5$ hours/week on Cello
- $x_6$ hours/week on Facebook
- $x_7$ hours/week sleeping

Objective

To maximise in this example, but for some other problems, it may be more appropriate to minimise

$\sum x_i \geq 11$  \quad $x_4 + x_7 \leq 5$  \quad $x_4 + x_8 < 50$

Decision variables

Constraints

Finding optimum solutions

- Solver allows users to solve constrained optimisation problems.
- Typically we want to maximise or minimise a value of a variable or a function. Such a function is called an objective function. For example, we may want to maximise profit or minimise time on a project.
- In many cases the solution space is huge, and solver cannot find the best solution. Generally we will accept a nearly-optimal solution.
- Solver uses the following inputs:
  - An objective function (a cell containing a formula)
  - A set of decision variables that can be changed
  - A set of constraints to be satisfied
Calc and Excel

- Constraint expressions must be in a single cell.
- Calc’s Solver is limited to linear equations only. Excel can do non-linear optimisation as well.
- Calc’s Solver doesn’t always find the optimal solution, Excel’s is more robust.
- There is a non-linear extension for Calc’s solver, but it converges very slowly, if at all, and isn’t really suitable for general use.
- We’ll limit the examples to linear ones, but the principles also apply to non-linear models, so you can apply the knowledge to Excel in later courses.

Example 1: Wyndor

- Classic linear programming (+ optimisation) problem, can be solved graphically or algorithmically
- The Wyndor Glass Co has three plants
  - Plant P1 makes aluminium frames
  - Plant P2 makes wood frames
  - Plant P3 produces glass
- There are two products, both in demand
  - Product 1: glass door with aluminium frame
  - Product 2: glass window with wood frame
- Each product spends a certain time in each plant, and has an associated profit (over)
- Goal: develop a production plan (how many to make of each product) to maximise profit

Wyndor data

<table>
<thead>
<tr>
<th>Plant</th>
<th>Hours per batch</th>
<th>Plant hours available</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>P2</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>P3</td>
<td>3</td>
<td>18</td>
</tr>
<tr>
<td>Profit per batch</td>
<td>$3,000</td>
<td>$5,000</td>
</tr>
</tbody>
</table>

Analysis

- Need to identify the objective function, decision variables and constraints
  - decision variables = production quantities (doors, windows)
  - objective = total profit = sum(quantity x profit per batch)
  - constraints = plant availability limits of the form
    - sum(quantity x hours in plant Pn) <= available hours in Pn
- Basic data goes on the sheet as a table
- Add cells for variables, align with columns
- Add cells for intermediate calculations subject to constraints
  - here it’s the total production time required at each plant
  - Colour coding is helpful:

Spreadsheet modelling

Solver data entry
Solver solution

<table>
<thead>
<tr>
<th>Plant</th>
<th>Hours per batch</th>
<th>Plant hours available</th>
<th>Plant hours required</th>
<th>All constraints satisfied (2 of 3 plants at capacity)</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>P2</td>
<td>2</td>
<td>2</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>P3</td>
<td>3</td>
<td>2</td>
<td>15</td>
<td></td>
</tr>
</tbody>
</table>

Profit per batch: $3,000
Production: 2
Total Profit: $30,000

Agrees with old-fashioned graphical solution: intersection of constraint regions and objective line on doors vs windows plot.

Example 2: Radiotherapy

- Calculate period over which different radiation beams are activated to minimise damage to healthy tissue while delivering sufficient dosage to tumour
  - decision variables = time for each beam
  - objective = dosage to healthy tissue, minimise
  - constraints = sufficient dosage to tumour centre and edge; upper limit to specially sensitive tissue

Example 2: Radiotherapy

- Calculate period over which different radiation beams are activated to minimise damage to healthy tissue while delivering sufficient dosage to tumour
  - decision variables = time for each beam
  - objective = dosage to healthy tissue, minimise
  - constraints = sufficient dosage to tumour centre and edge; upper limit to specially sensitive tissue

Battle of water networks

- Annual competition to improve the efficiency, safety, operation of water networks
- A different problem is studied each year

Problem: Detecting water contamination

- Contaminants can be injected into water distribution networks
  - Accidentally (e.g. spillage from factories)
  - Intentionally (e.g. terrorism)
- Need to put sensors in the water distribution networks to detect possible water contamination

Water network optimisation (1)

- Given: 5 possible locations to place sensors
- Decision variables:
  - $X_i$ = whether to put a sensor at possible location $i$
  - $X_i$ can be either 1 or 0
  - $X_i = 1$ means put a sensor at possible location $i$
  - $X_i = 0$ means do not put a sensor at possible location $i$
Water network optimisation (2)

- **Given:**
  - There are 5 regions: A, B, C, D and E
  - Each sensor can cover certain regions
- **Constraints:** Each region must be covered by at least one sensor
- **Given:** Each possible location has a deployment and maintenance cost
- **Objective:** Total cost
- **Optimisation:** Minimise total cost subject to constraints

Example 3: Water network

- Table with 1s shows which region is covered by which possible sensor location
- Allows use of `sumproduct` function to simplify formulas

=**sumproduct**(B9:F9;B10:F10)

Mixed ref formula can be filled down

Matrix arithmetic

- **Not for Semester 1**

Options and cautions

- This is an example of an integer problem, can’t have a fraction of a sensor (or in staff scheduling, a fraction of a person)
- Also can’t have negative sensors (or people)
- All Solvers have options, Calc’s is limited to
  - assume variables as integer
  - assume variables as non-negative
  - iteration limit if solution is elusive
- Regrettably, sometimes they aren’t always 100% effective
- Calc’s solver can get stuck in a local minimum for some scheduling problems, and finds a slightly sub-optimal solution

Matrix representation

- Calc and Excel provide standard matrix operations such as add, multiply, inverse, transpose, determinant, etc
- Calc and Excel use the term array to refer to a collection of values in a range that should be kept together.
- Both vectors and matrices are considered arrays.
- We can name an array, using Insert – Names – Define…, or by typing directly in the Name Box.
  - This makes the formulas much easier to enter and understand
  - Calc’s names are unique to the document, Excel’s can be local to the sheet or document-wide
- Need to use Ctrl-Shift-Enter to enter array formulas, or the Function Wizard (recommended)
  - formulas display (but aren’t entered) with curly brackets `{=A+B}`
Solving simultaneous equations

• Using Matrix operations:

\[
\begin{align*}
\text{Not for Semester 1} \\
{=} & \text{MINVERSE}(A) \\
{=} & \text{MMULT}(\text{MINVERSE}(A), b)
\end{align*}
\]

Financial functions

Excel offers a wide range of financial functions. For example:

• \text{PMT} (i; n; P)
  Returns the periodic (e.g. monthly) payment for an \(n\)-payment loan of \(P\) dollars at interest rate \(i\) per period. Sign is opposite to that of the principal.

• \text{PV} (i; n; A)
  Returns the present value of a series of \(n\) payments of \(A\) dollars each at interest rate \(i\) per period, given the intended final value.
  
  For this function, see Calc function list at https://wiki.openoffice.org/wiki/Documentation/How_Tos/Calc:_Financial_functions

Cross (data) tables

Cross Tables (Calc) or Data Tables (Excel):

– A cross/data table is a range of cells that shows how changing certain values in your formulas affects the results of the formulas.

– These tables provide a shortcut for calculating multiple versions in one operation and a way to view and compare the results of changing two variables together on your worksheet.

– See CrossTable sheet for an example with instructions

Lookup functions

\text{VLOOKUP} (search_value; table; column; lookup_type)

Searches for a value in the leftmost column of a table, and then returns a value in the same row.

– search_value can be a number or a string

– table is a range, can be on another sheet

– column is the column number from which the return value is obtained, relative to the search column (= 1)

– lookup_type is 1 (true) if the data is sorted on the first column, else 0

\text{HLOOKUP} works the same on rows

\text{MATCH} (search_value; vector; match_type)

Searches for a value in vector, a single row or column, returns position relative to start. match_type 0 means the data is unsorted, otherwise 1 = ascending, -1 = descending.

\text{INDEX} (vector; position)

Returns the value at position relative to the start of a vector

Scenarios

• Scenario:
  – A scenario contains a set of values that Calc or Excel saves and can substitute automatically in your worksheet.
  – You can create and save different groups of values on a worksheet with a comment, and then switch to any of these new scenarios to view different results.
  – Calc’s scenarios can be easily selected on screen
  – Example – financial or patient radiology parameters

Spreadsheet risks

• In spreadsheets, computational steps are often scattered across a wide range of cells and even across many sheets.

• It is very easy to make mistakes while developing a solution in Calc or Excel. Therefore, it is essential to thoroughly test your solutions.

$24-million spreadsheet “clerical error”

June 03, 2003 TORONTO (Reuters) - TransAlta Corp. said on Tuesday it will take a $24 million charge to earnings after a bidding snafu landed it more U.S. power transmission hedging contracts than it bargained for, at higher prices than it wanted to pay.

[...] the company’s computer spreadsheet contained mismatched bids for the contracts, it said. “It was literally a cut-and-paste error,” an Excel spreadsheet that we did not detect until we reviewed final sorting and ranking bids prior to submission,” TransAlta chief executive Steve Snyder said in a conference call. “I am clearly disappointed over this event. The important thing is to learn from it, which we’ve done.”
Spreadsheet errors

- From “How do you know your spreadsheet is right?: Principles, Techniques and Practice of Spreadsheet Style” by Philip L. Bewig, July 28, 2005 (available at [http://www.eusprig.org/hdykysir.pdf](http://www.eusprig.org/hdykysir.pdf)):
  - A missing minus sign caused Fidelity’s Magellan Fund to overstate projected earnings by $2.6 billion (yes, billion) and miss a promised dividend.
  - False-linked spreadsheets permitted fraud totaling $700 million at the Allied Irish Bank.
  - Voting officials reported spreadsheet irregularities in New Mexico and South Africa.
- More examples (again a few years old) from the same source European Spreadsheet Risks Interest Group (yes, the organisation does exist): [http://www.eusprig.org/stories.htm](http://www.eusprig.org/stories.htm)
- Survey of error rates in spreadsheets: [http://panko.shidler.hawaii.edu/ssr/Mypapers/whatknow.htm](http://panko.shidler.hawaii.edu/ssr/Mypapers/whatknow.htm)

Avoiding errors 1

Strategies for avoiding making mistakes (or quickly finding any you do make) vary according to how you’re using the tool:

- If the sheet is used to conduct a straightforward analysis of a lot of data,
  - Test the analysis with a small set of data for which you know the answers
  - Examine the real data for evidence of corruption (missing or misinterpreted values)
  - Make use of spreadsheet features that can highlight unusual or erroneous features in the data (remember the geyser interval quantisation, only discovered when the data was sorted)
  - Conditional formatting and statistical functions are your friends
  - Check for normally invisible changes like formulas accidentally replaced by values or text/number mixups
- Turn on View – Value Highlighting temporarily, use it to find an error of each of these kinds on the Sturec sheet

Avoiding errors 2

- If the sheet contains a model of some process, with input measurements and corresponding outputs from the model
  - Design the sheet to show the process flow and logical grouping of elements
    - Label the cells clearly
    - Use cell colour to code meaning
    - At least clearly separate inputs and outputs
  - Use simple inputs first, compare results with the expected answers, gradually build up to the real input values
  - If results are unexpected, check cell references in formulas (could be incorrect usage of relative/mixed/absolute addressing exposed by copy/paste or fill actions)
  - Tools – Detective might be an overblown name, but it can be handy to see which cells depend on which others

Epilogue

- This is the formal end of the SS topic.
- Next week’s lab will cover Solver, pivot tables from last week, a financial calculator and a find-the-data-errors puzzle.
- Skills developed in labs 02, 03 and 04 are assessed in the midterm practical test in week 5.
- If there’s time we’ll finish off with an example showing the development of a solution from scratch, either
  - finding a data set, loading into Calc, devising the analysis and testing and finalisation
  - identifying a problem, establishing a model that can produce a solution, devising the sheet layout, testing and finalisation