Advanced Business Modeling

CIS 418

Linear Programming: Production problems

Why Linear Programming?

- Linear programming (**LP**) is a tool to solve decision problems where
  - **relationships** between **decision variables** and the **objective** are **linear**
  - **relationship**s between **decision variables** and **constraints** are **linear**

- Widely used in for a variety of applications. For example:
  - production planning
  - Portfolio optimization
  - Scheduling
- Algorithms exist to find globally optimal solutions

Linear vs. Non-linear

- Linear: sum of variables, multiple by scalar (coefficient)

$$x_1$$

$$5x_1$$

$$5x_1 + 3x_2$$

$$5x_1 + 3x_2 + x_3 + 100x_4 + 2.33x_5 + \ldots$$

- Non-linear: x_1^2 $\sqrt[3]{x_1}$ $\ln(x_1)$ $\sin(x_1)$ e^{x_1} $5x_1 + \ln(x_2)$ $x_1 \cdot x_2$

This formulation is linear

$x_1 + x_3 \geq x_2$

But this one is not

$(x_1+x_3)/x_2 \geq 1$

LP: constraints and objective are linear functions

Example: Desks or Tables?

A manufacturer makes wooden desks and tables.

Each desk requires 4 hours to cut and 2 hours to assemble.

Each table requires 3 hours to cut and 5 hours to assemble.

The manufacturer can do only up to 12 hours of cutting and 10 hours of assembling per day.

Profit is 15$ per desk and 12$ per table.

What would be the production plan that maximizes the profit?

Formulate the problem

Variables: Desks -X_1 Tables -X_2

Constraints:

Cutting: $4x_1 + 3x_2 \le 12$

Assembling: $2x_1 + 5x_2 \le 10$

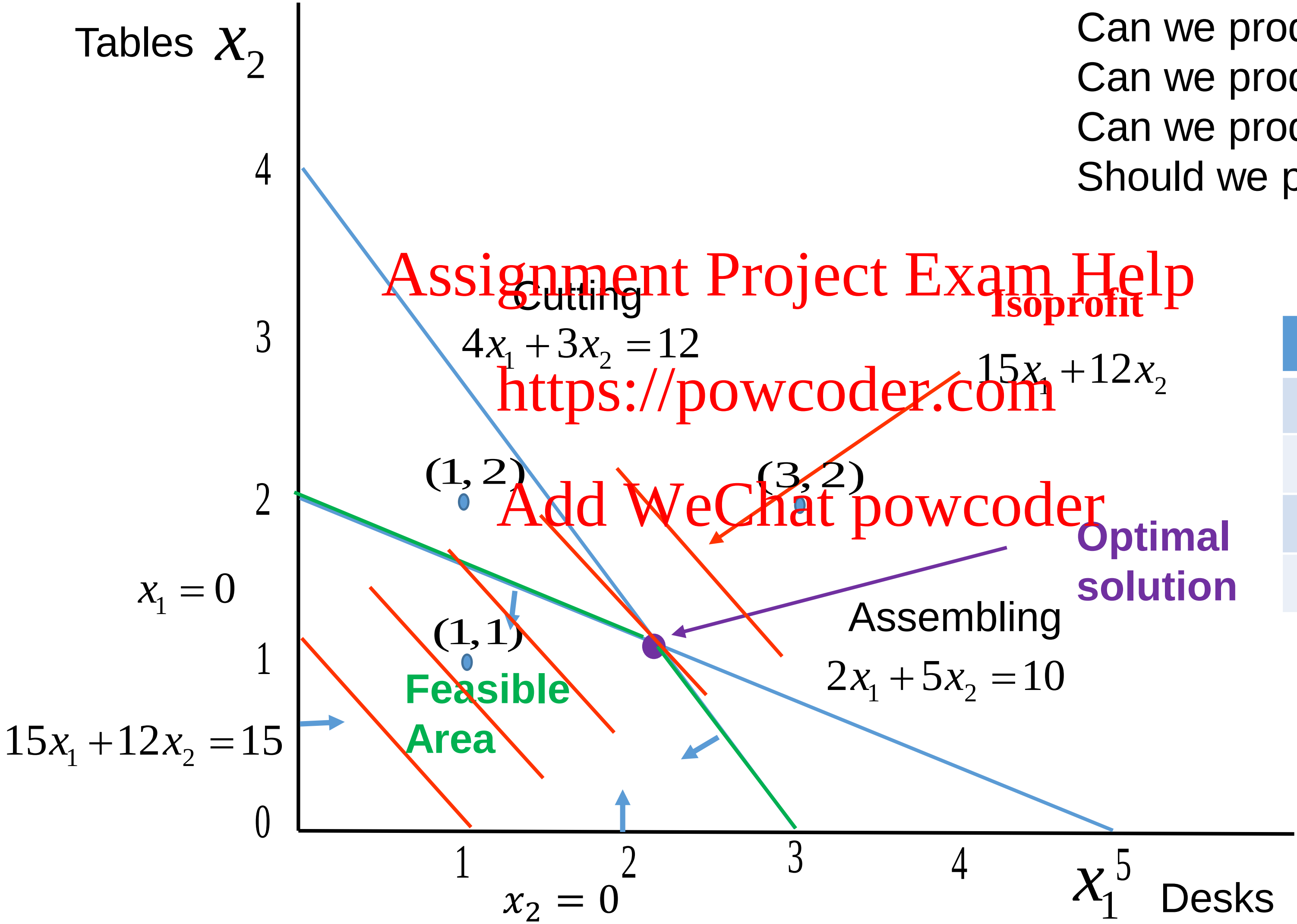
Linear

Non-negative production: $x_1 \ge 0, \quad x_2 \ge 0$

Objective: max profit $15x_1 + 12x_2$ Linear

Graphic Solution

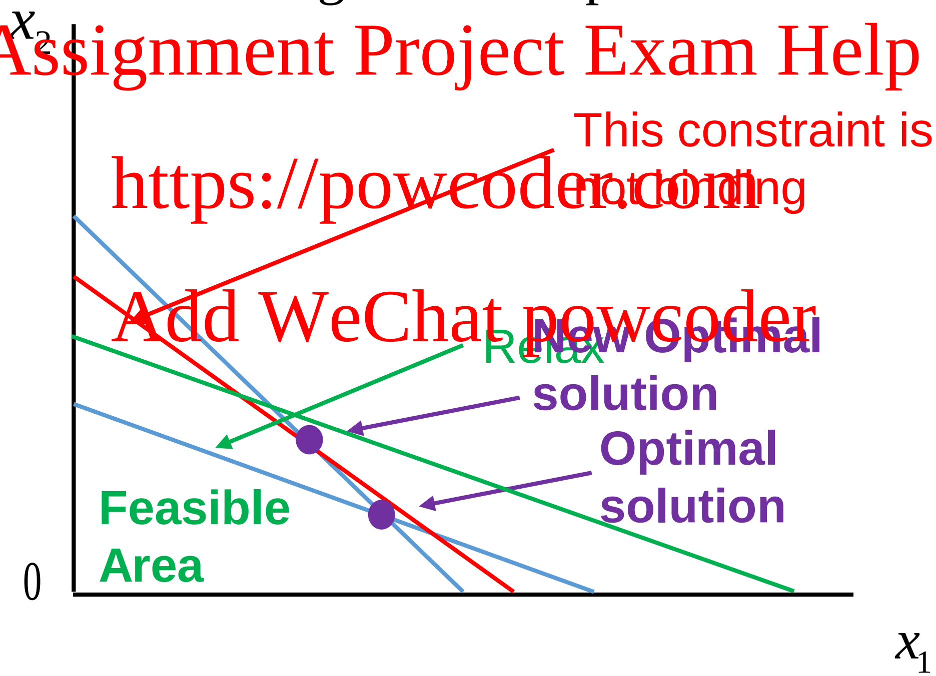
Can we produce (3,2)?
Can we produce (1,2)?
Can we produce (1,1)?
Should we produce (1,1)?



Nodes	Profit
(0,0)	0
(0,2)	24
(3,0)	45
(2.14,1.14)	45.78

Feasible area and optimal solution

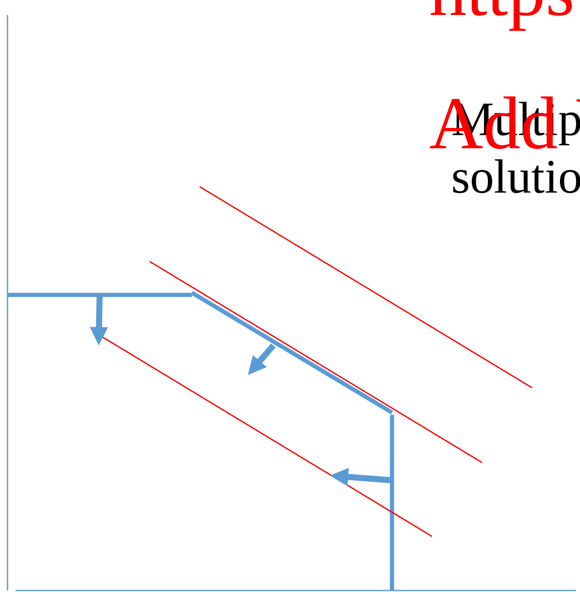
- Feasible area – the set of all possible variable combinations.
- Binding constraints – constraints that define the edges of the feasible set.
- Not all constraints are binding. For example:

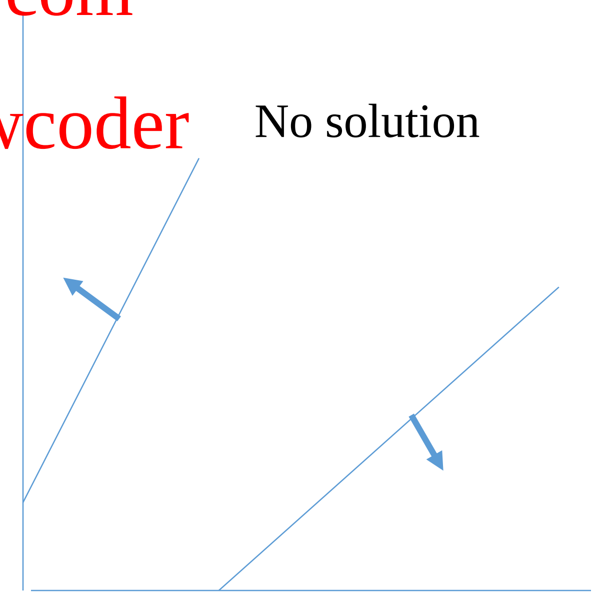


- If we relax a constraint (if we have more resources), the feasible area might change and so is the solution.

Solution properties

- An LP problem could have either:
  - A unique solution – only if the feasible set is bounded and has no holes.
  - Multiple solutions – if the isoprofit line is parallel to a binding constraint line.
  - No solution – for example, if the feasible set is not bounded, or if the feasible set is null.





Solution properties - Summary

- Isoprofit line – the line that represents the objective function.
- The isoprofit lines are parallel to each other
- A unique solution would be found on the intersection between binding constraints.
- The solution can change if:
  - A binding constraint is relaxed
  - Another constraint is added
  - The profitability ratios between variables change (and therefore the objective function changes).

Example: Go-Green!

- **Go-Green!** company provides all the ingredients plus the recipes for cooking your own meals at home. Customers can choose between Organic, Vegan and Green meals.
- Each meal is built from a mix of ingredients in inventory:
  - Peppers
  - Kale
  - Tomatoes
  - Butternut squash
  - Arugula
- Your goal is to produce the mix of meals which will maximize profits, given the inventory.

All parameters are detailed in the excel file LP_problems.xlsx

Formulate the problem

1. What are the **variables**? Name them $(x_1, x_2, \ldots)$

2. What are the **constraints**? Write them in a mathematical form (inequalities)

3. What is the **objective function**? Write it in a mathematical form.

4. Do we **maximize** or **minimize** the objective?

Formulate the problem

1. What are the **variables**? Name them $(x_1, x_2, ...)$

Organic- x_1 Vegan- x_2 Green- x_3

2. What are the **constraints**? Write them in a mathematical form (inequalities)

- Peppers $x_1 + x_2 \leq 450$
- Kale $x_1 \leq 250$
- Tomatoes $2x_1 + 2x_2 + x_3 \leq 800$
- Butternut squash $x_1 + x_2 \leq 450$
- Arugula $2x_1 + x_2 + x_3 \leq 600$
- Non-negative production: $x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0$

3. What is the **objective function**? Write it in a mathematical form.

Profit: $75x_1 + 50x_2 + 35x_3$

4. Do we **maximize** or **minimize** the objective? **Maximize**

Feasibility

- Build a **spreadsheet** to **calculate the profit** from a mix of
  - ✓200 Organic meals,
  - ✓200 Vegan meals, and
  - ✓10 Green meals.

- Is it **feasible** to build this combination of products? How do you know?

Feasibility check

PARAMETERS

	Profit Margin
Organic dinner	$ 75.00
Vegan dinner	$ 50.00
Green dinner	$ 35.00

Bill of Materials

	Peppers	Kale	Tomatoes	Butternut squash	Arugula
Organic dinner	1	1	2	1	2
Vegan dinner	1	0	2	1	1
Green dinner	0	0	1	0	1

Inventory

Peppers	Kale	Tomatoes	Butternut squash	Arugula
450	250	800	450	600

Decision Variables

Organic dinner	200
Vegan dinner	200
Green dinner	10

Objective

	Profit
Organic dinner	$ 15,000.00
Vegan dinner	$ 10,000.00
Green dinner	$ 350.00
Total Profit	$ 25,350.00

Calculations

	Used		Avaliable
Peppers	400	<	450
Kale	200	<	250
Tomatoes	810	>	800
Butternut squash	400	<	450
Arugula	610	>	600

Solving the optimization problem

- The problem has more than two variables – therefore it is more difficult to solve on paper (we would need to draw a 3D graph…)
- Therefore we solve it by using Excel Solver.

When we solve an optimization problem using excel, we do the following:

- Choose arbitrary variables (Guess!)
- Calculate the constraints using those arbitrary variables

- Calculate the objective using those arbitrary variables
- Use solver and define the variables, objective and constraints

- Let solver find the solution for you

LP using vector and matrix notation

n decision variables as a **column vector**

$$\begin{bmatrix} x_1 \\ x_2 \\ \\ x_n \end{bmatrix}$$

Product of a **row vector** and a **column vector** as the objective

$$\begin{bmatrix} c_1 & c_2 & & c_n \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ \\ x_n \end{bmatrix} = c_1 x_1 + c_2 x_2 + \cdots + c_n x_n$$

m constraints, all constraints are linear combinations of the *n* decision variables

$$\begin{bmatrix} a_{11} & a_{12} & & a_{1n} \\ a_{21} & a_{22} & & a_{2n} \\ \\ a_{m1} & a_{m2} & & a_{mn} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ \\ x_n \end{bmatrix} = \begin{bmatrix} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n \\ \\ a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n \end{bmatrix} \le \begin{bmatrix} b_1 \\ b_2 \\ \\ b_m \end{bmatrix}$$

Very brief review of matrix (array) multiplication

Array dimensions are denoted as [**number of rows** x **number of columns**]

[**n** x 1] [1 x **n**] [**m** x **n**]

$$\begin{bmatrix} x_1 \\ x_2 \\ \\ x_n \end{bmatrix} \qquad \begin{bmatrix} c_1 & c_2 & & c_n \end{bmatrix} \qquad \begin{bmatrix} a_{11} & a_{12} & & a_{1n} \\ a_{21} & a_{22} & & a_{2n} \\ & & & \\ a_{m1} & a_{m2} & & a_{mn} \end{bmatrix}.$$

Two matrices can be multiplied if **their internal dimensions match**. The result has the **same number of rows as the first** matrix and the same **number of columns as the second**.

[1 x **n**] x [**n** x 1] = [1 x 1]

$$\begin{bmatrix} c_1 & c_2 & \cdots & c_n \end{bmatrix} \times \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = c_1 x_1 + c_2 x_2 + \cdots + c_n x_n$$

A very brief review of how to multiply matrices

Matrix multiplication is not commutative: AxB ≠ BxA

[**n** x 1] x [1 x **n**] = [**n** x **n**]

$$\begin{bmatrix} x_1 \\ x_2 \\ \\ x_n \end{bmatrix} \times \begin{bmatrix} c_1 & c_2 & & c_n \end{bmatrix} = \begin{bmatrix} x_1c_1 & x_1c_2 & & x_1c_n \\ x_2c_1 & x_2c_2 & & x_2c_n \\ & & & \\ x_nc_1 & x_nc_2 & & x_nc_n \end{bmatrix}$$

[1 x **n**] x [**n** x 1] = [1 x 1]

$$\begin{bmatrix} c_1 & c_2 & & c_n \end{bmatrix} \times \begin{bmatrix} x_1 \\ x_2 \\ \\ x_n \end{bmatrix} = c_1x_1 + c_2x_2 + \cdots + c_nx_n$$

The element in row **r** and column **m** of the **result** is the **matrix product** of

- **row r** of the **first matrix** and
- **column m** of the **second matrix.**

Useful excel functions

- SUMPRODUCT – sum the product of two arrays.
  The dimension of the two arrays must be equal!
  - Array1 = [1 2 3 5]
  - Array2 = [4 5 6 2]

  Sumproduct(Array1, Array2)=1*4+2*5+3*6+5*2=42

- TRANSPOSE – transpose a matrix.
  - Array1 = [1 2 3 5] TRANSPOSE(Array1) = $\begin{bmatrix} 1 \\ 2 \\ 3 \\ 5 \end{bmatrix}$

  - Matrix2 = $\begin{bmatrix} 3 & 2 & 1 & 7 \\ 2 & 4 & 5 & 1 \\ 6 & 5 & 2 & 1 \end{bmatrix}$ TRANSPOSE(Matrix2) = $\begin{bmatrix} 3 & 2 & 6 \\ 2 & 4 & 5 \\ 1 & 5 & 2 \\ 7 & 1 & 1 \end{bmatrix}$

If you are using the TRANSPOSE function, you must enter the formula in as an array formula, i.e. use **<CTRL><SHIFT><ENTER>**

Useful excel functions (cont.)

- Mmult – the product of two matrices.

  The inner dimensions of the two matrices must be equal!

  Matrix1: [5 3 2] - it is a 1x3 matrix

  Matrix2: $\begin{bmatrix} 3 & 2 & 1 & 7 \\ 2 & 4 & 5 & 1 \\ 6 & 5 & 2 & 1 \end{bmatrix}$ - it is a 3x4 matrix

Mmult(Matrix1, Matrix2)= [33 32 24 40] -1x4 matrix (=vector in this case)

- For array operations the first step is to **select the area** that will have the **correct dimensions** for the **result.**
- After entering the function use **<u><CTRL><SHIFT></u><ENTER>** instead of just **<ENTER>** to let Excel know that you want to use an array function. You will see curly brackets around your function.
- Use MMULT(**A,B**) to multiply two matrices: **A** and **B**. Remember MMULT(**A,B**)≠ MMULT(**B,A**). To multiply more than two matrices, you can nest MMULT() operations.

Setting up decision variables as a row vector

PARAMETERS

	Profit Margin
Organic dinner	$ 75.00
Vegan dinner	$ 50.00
Green dinner	$ 35.00

Bill of Materials

	Peppers	Kale	Tomatoes	Butternut squash	Arugula
Organic dinner	1	1	2	1	2
Vegan dinner	1	0	2	1	1
Green dinner	0	0	1	0	1

Inventory

Peppers	Kale	Tomatoes	Butternut squash	Arugula
450	250	800	450	600

Objective: profit

$$[x_{Organic} \quad x_{Vegan} \quad x_{Green}] \times \begin{bmatrix} 75 \\ 50 \\ 35 \end{bmatrix} = 75x_{Organic} + 50x_{Vegan} + 35x_{Green}$$

Constraints: inventory

$$[x_{Organic} \quad x_{Vegan} \quad x_{Green}] \times \begin{bmatrix} 1 & 1 & 2 & 1 & 2 \\ 1 & 0 & 2 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \end{bmatrix} =$$

$$= \quad [(x_{Or} + x_V) \quad (x_{Or}) \quad (2x_{Or} + 2x_V + x_G) \quad (x_{Or} + x_V) \quad (2x_{Or} + x_V + x_G)]$$

Setting up decision variables as a column vector

PARAMETERS

	Profit Margin
Organic dinner	$ 75.00
Vegan dinner	$ 50.00
Green dinner	$ 35.00

Bill of Materials

	Peppers	Kale	Tomatoes	Butternut squash	Arugula
Organic dinner	1	1	2	1	2
Vegan dinner	1	0	2	1	1
Green dinner	0	0	1	0	1

Inventory

Peppers	Kale	Tomatoes	Butternut squash	Arugula
450	250	800	450	600

Objective: profit

$$[75 \quad 50 \quad 35] \times \begin{bmatrix} x_{Organic} \\ x_{Vegan} \\ x_{Green} \end{bmatrix} = 75x_{Organic} + 50x_{Vegan} + 35x_{Green}$$

Constraints: inventory

$$\begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 0 \\ 2 & 2 & 1 \\ 1 & 1 & 0 \\ 2 & 1 & 1 \end{bmatrix} \times \begin{bmatrix} x_{Organic} \\ x_{Vegan} \\ x_{Green} \end{bmatrix} = \begin{bmatrix} x_{Organic} + x_{Vegan} \\ x_{Organic} \\ 2x_{Organic} + 2x_{Vegan} + x_{Green} \\ x_{Organic} + x_{Vegan} \\ 2x_{Organic} + x_{Vegan} + x_{Green} \end{bmatrix}$$

Robustness

- Problem: if something changes in the data, the solution might change
- If you are running a business, you want to make sure that small changes in prices would not drag you into a completely different production setting.
- Would the current solution still be optimal?
- How robust is the solution?

- Answer: by sensitivity analysis.
- Go back to the spreadsheet, in risk solver choose

Reports ->Optimization -> Sensitivity

Sensitivity Report

Objective Cell (Max)

Cell	Name	Final Value
G21	Total Profit	25000

Decision Variable Cells

Cell	Name	Final Value	Reduced Cost	Objective Coefficient	Allowable Increase	Allowable Decrease
C18	Organic dinner Profit Margin	200	0	75	25.0000002	5.0000002
C19	Vegan dinner Profit Margin	200	0	50	25.0000001	12.5000001
C20	Green dinner Profit Margin	0	-2.5	35	2.5	1E+30

Constraints

Cell	Name	Final Value	Shadow Price	Constraint R.H. Side	Allowable Increase	Allowable Decrease
G26	Peppers	400	0	450	1E+30	50
H26	Kale	200	0	250	1E+30	50
I26	Tomatoes	800	12.5	800	100	100
J26	Butternut squash	400	0	450	1E+30	50
K26	Arugula	600	25	600	50	200

What happens if the profit margins change?

Decision Variable Cells

Cell	Name	Final Value	Reduced Cost	Objective Coefficient	Allowable Increase	Allowable Decrease
C18	Organic dinner Profit Margin	200	0	75	25.0000002	5.0000002
C19	Vegan dinner Profit Margin	200	0	50	25.0000001	12.5000001
C20	Green dinner Profit Margin	0	-2.5	35	2.5	1E+30

These capture a change in the isoprofit function.

As long as profit margins stay in the allowable region, the optimal solution does not change.

If the Organic dinner profit margin rose up to $100 per dinner or fell to $70 per dinner (while other profit margins did not change), it would still be optimal to produce 200 Organic dinners and 200 Vegan dinners.

The optimal mix would change if the Organic dinner profit margin increased past $100, or decreased below $70.

What is the reduced cost?

Decision Variable Cells

Cell	Name	Final Value	Reduced Cost	Objective Coefficient	Allowable Increase	Allowable Decrease
C18	Organic dinner Profit Margin	200	0	75	25.0000002	5.0000002
C19	Vegan dinner Profit Margin	200	0	50	25.0000001	12.5000001
C20	Green dinner Profit Margin	0	-2.5	35	2.5	1E+30

Holding all else constant, the profit margin for Green dinners will have to increase by $2.50 before we would consider producing any Green dinners out of our inventory.

Shadow prices

Non-zero shadow prices (aka dual variables) are associated with binding constraints.

Constraints

Cell	Name	Final Value	Shadow Price	Constraint R.H. Side	Allowable Increase	Allowable Decrease
G26	Peppers	400	0	450	1E+30	50
H26	Kale	200	0	250	1E+30	50
I26	Tomatoes	800	12.5	800	100	100
J26	Butternut squash	400	0	450	1E+30	50
K26	Arugula	600	25	600	50	200

Just ONE more tomato would increase profits by $12.50, one extra arugula would increase profits by $25.

If the supply of arugula increases past 650 units, or decreases past 400 units, the shadow price associated with the arugula changes.

Shadow prices

Lets assume we do not have the sensitivity report.
How can we figure out the shadow price of tomatoes?

Shadow prices

Lets assume we do not have the sensitivity report.
How can we figure out the shadow price of tpmatoes?

__Solution__:

- We add one more tomato to the inventory.
- We keep everything else fixed.
- We calculate the optimal profit with the additional tomato.
- The shadow price of a tomato is the difference between the new profit and the previous profit.
- If the profit did not change the price would be 0. It happens if the constraint over tomatoes is not binding.