

生産計画をたてる

材料\製品	α	β	γ	δ	ε	量
A	3	0	1	3	1	80
B	1	2	0	3	2	75
C	0	4	2	5	0	95
D	2	1	0	1	2	70
利益	6	7	3	10	5	

➤ 生産計画の最適化(例1)

- 4種の材料 A, B, C, D がある
- 5種の製品 $\alpha, \beta, \gamma, \delta, \varepsilon$ をつくる
- 各材料の所有数, 各製品を1単位作るのに必要な材料数, 利益は右表

➤ **総利益を最大**にする生産計画をたてたい

➤ 最適化問題の定式化(変数設定)

- 変数 x_i ... 製品 i の作成数(単位)
- 変数ベクトル $x = (x_1, x_2, x_3, x_4, x_5)$

➤ 最適化問題の定式化(係数表記)

- 利益を表す係数ベクトル c
- 材料所持数を表す係数ベクトル b
- 必要材料数を表す係数行列 A

$$\begin{aligned}
 c &= (c_1 \quad c_2 \quad c_3 \quad c_4 \quad c_5) \\
 &= (6 \quad 7 \quad 3 \quad 10 \quad 5) \\
 b &= (b_1 \quad b_2 \quad b_3 \quad b_4) \\
 &= (80 \quad 75 \quad 95 \quad 70) \\
 A &= \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} \\ a_{21} & a_{22} & a_{23} & a_{24} & a_{25} \\ a_{31} & a_{32} & a_{33} & a_{34} & a_{35} \\ a_{41} & a_{42} & a_{43} & a_{44} & a_{45} \end{pmatrix} \\
 &= \begin{pmatrix} 3 & 0 & 1 & 3 & 1 \\ 1 & 2 & 0 & 3 & 2 \\ 0 & 4 & 2 & 5 & 0 \\ 2 & 1 & 0 & 1 & 2 \end{pmatrix}
 \end{aligned}$$

生産計画をたてる

➤ 最適化問題の定式化(ベタ表記)

$$\begin{aligned} \min. & \quad 6x_1 + 7x_2 + 3x_3 + 10x_4 + 5x_5 \\ \text{s. t.} & \quad 3x_{11} + 0x_{12} + 1x_{13} + 3x_{14} + 1x_{15} \leq 80 \\ & \quad 1x_{21} + 2x_{22} + 0x_{23} + 3x_{24} + 2x_{25} \leq 75 \\ & \quad 0x_{31} + 4x_{32} + 2x_{33} + 5x_{34} + 0x_{35} \leq 95 \\ & \quad 2x_{41} + 1x_{42} + 0x_{43} + 1x_{44} + 2x_{45} \leq 70 \\ & \quad x_{11}, x_{12}, x_{13}, x_{14}, x_{15}, \dots, x_{45} \geq 0 \end{aligned}$$

$$c = (c_1 \quad c_2 \quad c_3 \quad c_4 \quad c_5)$$

$$= (6 \quad 7 \quad 3 \quad 10 \quad 5)$$

$$b = (b_1 \quad b_2 \quad b_3 \quad b_4)$$

$$= (80 \quad 75 \quad 95 \quad 70)$$

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} \\ a_{21} & a_{22} & a_{23} & a_{24} & a_{25} \\ a_{31} & a_{32} & a_{33} & a_{34} & a_{35} \\ a_{41} & a_{42} & a_{43} & a_{44} & a_{45} \end{pmatrix}$$

$$= \begin{pmatrix} 3 & 0 & 1 & 3 & 1 \\ 1 & 2 & 0 & 3 & 2 \\ 0 & 4 & 2 & 5 & 0 \\ 2 & 1 & 0 & 1 & 2 \end{pmatrix}$$

➤ 最適化問題の定式化(Σ 表記)

$$\min. \sum_{j=1}^5 c_j x_j$$

$$\text{s. t. } \sum_{j=1}^5 a_{ij} x_j \leq b_i \quad (i = 1, \dots, 4)$$

$$x_j \geq 0 \quad (j = 1, \dots, 5)$$

How to use CPLEX?

➤ 新規プロジェクトの作成

- ① [ファイル(F)]ー[新規(N)]ー[OPLプロジェクト]を選択
- ② [プロジェクト名]を記入(例: **ProductionPlanning**)し, 3カ所にチェックする
 - デフォルトの実行構成の追加
 - モデルの作成
 - データの作成
- ③ [終了]をクリック

プロジェクト名は自由だが, **半角英数**で何の問題を解こうとしているのかが分かる名前が良い

➤ プロジェクト内のいくつかの名前を変更

- ✓ [構成1] → [**config1**] ※日本語を英語に変更しないと実行時エラーになる
- ✓ モデルファイル [ProductionPlanning.mod] → [**pp.mod**]
- ✓ データファイル [ProductionPlanning.dat] → [**ppex1.dat**]

➤ モデルファイル・データファイルを記述し保存(次ページ参照)

➤ [config1]にモデルファイルとデータファイルをセットする

How to use CPLEX?

➤ pp.mod

```
int i_max = ...; // 材料数
int j_max = ...; // 製品数

range I = 1..i_max; // 材料集合の範囲
range J = 1..j_max; // 製品集合の範囲

int a[I,J] = ...; // 各製品1単位あたりの必要材料数を表す行列
int b[I] = ...; // 材料i の所持数
int c[J] = ...; // 製品j の1単位あたり利益

dvar float+ x[J]; // 製品j の生産量(非負)

maximize
    sum(j in J) c[j]*x[j];
subject to {
    forall(i in I) {
        sum(j in J) a[i,j]*x[j] <= b[i];
    }
}
```

How to use CPLEX?

➤ ppex1.dat

```
i_max = 4;  
j_max = 5;  
  
c = [6 7 3 10 5];  
b = [80 75 95 70];  
  
a = [  
[3 0 1 3 1]  
[1 2 0 3 2]  
[0 4 2 5 0]  
[2 1 0 1 2]  
];
```

How to use CPLEX?

➤ 計算結果の確認([解]タブの中身)

最適値

```
// solution (optimal) with objective 316
// Quality There are no bound infeasibilities.
// There are no reduced-cost infeasibilities.
// Maximum Ax-b residual           = 0
// Maximum c-B'pi residual         = 0
// Maximum |x|                     = 45.5
// Maximum |slack|                 = 4
// Maximum |pi|                    = 2.2
// Maximum |red-cost|              = 0.2
// Condition number of unscaled basis = 1.2e+01
//
```

```
x = [0 1 45.5 0 34.5];
```

optimal solution
最適解

生産計画をたてる

➤ 生産計画の最適化(例2)

- 7種の材料 A, B, ..., G がある
- 5種の製品 $\alpha, \beta, \gamma, \delta, \varepsilon$ をつくる
- 各材料の所有数, 各製品を1単位作るのに必要な材料数, 利益は右表
- **総利益最大**の生産計画をたてよ

材料\製品	α	β	γ	δ	ε	量
<i>A</i>	2	0	1	3	1	80
<i>B</i>	3	1	0	3	1	75
<i>C</i>	0	2	3	5	0	95
<i>D</i>	3	2	0	1	2	70
<i>E</i>	1	0	3	0	3	60
<i>F</i>	0	3	0	2	4	80
<i>G</i>	1	5	0	4	1	90
利益	8	7	3	9	6	

➤ 線形最適化問題として定式化し, CPLEXで解く

- モデルファイル[pp.mod]は共通なので, データファイル[ppex2.dat]を作り解く