

使用 Optimization Toolbox™ 进行基于问题的优化

使用自然语法定义和求解线性及混合整数线性、二次、线性最小二乘法及非线性优化问题。

1. 定义问题

按照 [基于问题的 workflows](#)，首先使用 `optimproblem` 创建优化问题，将目标、约束和关联变量包含在内。

示例：

```
assignmentProb = optimproblem
responseProb = optimproblem
```

2. 定义变量

使用 `optimvar` 创建优化变量。设置显示名称及可选尺寸、边界和类型。使用整数或字符串编制索引。

示例：

```
x = optimvar("x");
y = optimvar("y");

employees = ["a","b","c"];
tasks = ["t1","t2","t3"];
assign = optimvar("assign",employees,tasks,"LowerBound",0,"UpperBound",1,"Type","integer")
```

3. 定义要在目标和约束中使用的表达式

直接指定多项式比值 `OptimizationExpression`。

示例：

```
response = -3*(y - x.^3 - x).^2 - (x - 4/3).^2;
totalCost = sum(sum(cost.*assign));
sumByEmployee = sum(assign,2);
sumByTask = sum(assign,1);
```

将其他表达式指定为 MATLAB 函数，并使用 `fcn2optimexpr` 转换为优化表达式。

示例：

```
a = 4;
xyfcn = @(x,y,a)exp(y)*a*x.^2;
xyexpr = fcn2optimexpr(xyfcn,x,y,a);
```

4. 定义目标

设置优化 *偏好*。使用标量 `OptimizationExpression` 设置 *目标函数*。

示例：

```
responseProb.ObjectiveSense = "maximize";
responseProb.Objective = response;

assignmentProb.ObjectiveSense = "minimize";
assignmentProb.Objective = totalCost;
```

5. 定义约束

将 **OptimizationExpression** 与关系运算符结合使用来指定 **OptimizationConstraint**。分配给某个问题。

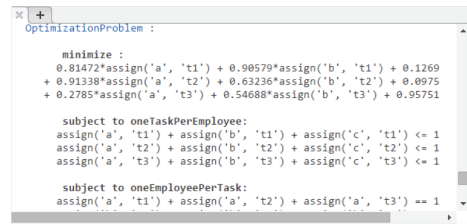
示例:

```
responseProb.Constraints.ellipse = x.^2/2 + y.^2/4 <= 1;
responseProb.Constraints.xyconstr = xyexpr >= 1;

assignmentProb.Constraints.oneTaskPerEmployee = sumByTask <= 1;
assignmentProb.Constraints.oneEmployeePerTask = sumByEmployee == 1;
```

6. 复习

使用 **showexpr**、**showconstr**、**showbounds** 和 **showproblem** 显示。

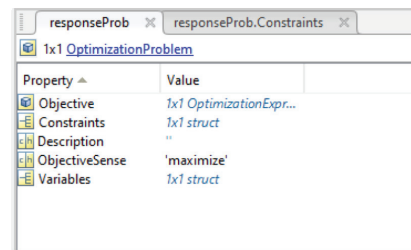


```
OptimizationProblem :
  minimize :
    0.81472*assign('a', 't1') + 0.90579*assign('b', 't1') + 0.1269
    + 0.91338*assign('a', 't2') + 0.63236*assign('b', 't2') + 0.0975
    + 0.2785*assign('a', 't3') + 0.54688*assign('b', 't3') + 0.95751

  subject to oneTaskPerEmployee:
    assign('a', 't1') + assign('b', 't1') + assign('c', 't1') <= 1
    assign('a', 't2') + assign('b', 't2') + assign('c', 't2') <= 1
    assign('a', 't3') + assign('b', 't3') + assign('c', 't3') <= 1

  subject to oneEmployeePerTask:
    assign('a', 't1') + assign('a', 't2') + assign('a', 't3') == 1
```

使用工作区浏览器查看。



Property	Value
Objective	1x1 OptimizationExpr...
Constraints	1x1 struct
Description	"
ObjectiveSense	'maximize'
Variables	1x1 struct

7. 求解和分析

使用 **Solve** 求解问题，返回解值、目标值及求解器停止的原因。
提供非线性问题的 **初始点**。

示例:

```
x0.x = 0;
x0.y = 0;
[sol,fval,exitflag] = solve(responseProb,x0)
```

```
sol = struct with fields:
  x: 0.8883
  y: 1.5563
  fval: -0.2013
  exitflag =
    OptimalSolution
```

使用 **优化选项** 求解。

示例:

```
o = optimoptions(assignProb,"MaxTime",10);
sol = solve(assignmentProb,"Options",o)
```

更多功能

- 使用 **evaluate** 和 **infeasibility** 分析结果
- **解释并改进** 结果
- 使用 **prob2struct** 转换为基于求解器的形式
- **包括导数**

了解更多: mathworks.com/help/optim