Different Types of LP Solutions

Alireza Sheikh-Zadeh, Ph.D.

Four different types of LP outcomes

1) a unique optimal solution
2) alternative (multiple) optimal solution
3) unbounded solution
4) infeasibile

Example: A model with unique optimal solution.

```
# Example: WYNDOR GLASS CO. (Maximization)
# Max: Z = 3x_1 + 5x_2
# S.t.
#  x_1        <= 4
#        2x_2 <= 12
# 3x_1 + 2X_2 <= 18
# X_1 and X_2 >= 0

# Model in Python
from pulp import *
# Define the Model
prob = LpProblem("WYNDOR GLASS CO.", LpMaximize)
x1 = LpVariable('x_1', lowBound = 0)
x2 = LpVariable('x_2', lowBound = 0)
# Objective function
prob += 3*x1 + 5*x2, "Obj"
# Constraints
prob += x1 <= 4
prob += 2*x2 <= 12
prob += 3*x1 + 2*x2 <= 18


print(prob)

## WYNDOR GLASS CO.:
## MAXIMIZE
## 3*x_1 + 5*x_2 + 0
## SUBJECT TO
## _C1: x_1 <= 4
##
## _C2: 2 x_2 <= 12
##
## _C3: 3 x_1 + 2 x_2 <= 18
##
## VARIABLES
```

```
## x_1 Continuous
## x_2 Continuous

prob.solve()

## 1

print("status: " + LpStatus[prob.status])

## status: Optimal

for variable in prob.variables():
     print("{}* = {}".format(variable.name, variable.varValue))


## x_1* = 2.0
## x_2* = 6.0

print(value(prob.objective))

# **Conclusions**

# The OR team used this approach to find that the optimal solution is $x_1 =
2$, $x_2 = 6$, with $Z = 36$. This solution indicates that the Wyndor Glass
Co. should produce products 1 and 2 at the rate of 2 batches per week and 6
batches per week, respectively, with a resulting total profit of $36,000 per
week. No other mix of the two products would be so profitable *according to
the model*.

## 36.0
```

Example: A model with alternative optimal solution.

```
# Example: Multiple Optimal Solution.
# Max: Z = 6x_1 + 4x_2
# S.t.
#  x_1        <= 4
#        2x_2 <= 12
# 3x_1 + 2X_2 <= 18
# X_1 and X_2 >= 0

# Model in Python
from pulp import *
# Define the Model
prob = LpProblem("Profit", LpMaximize)
x1 = LpVariable('x_1', lowBound = 0)
x2 = LpVariable('x_2', lowBound = 0)
# Objective function
prob += 6*x1 + 4*x2, "Obj"
# Constraints
prob += x1 <= 4
prob += 2*x2 <= 12
```

```
prob += 3*x1 + 2*x2 <= 18
print(prob)

## Profit:
## MAXIMIZE
## 6*x_1 + 4*x_2 + 0
## SUBJECT TO
## _C1: x_1 <= 4
##
## _C2: 2 x_2 <= 12
##
## _C3: 3 x_1 + 2 x_2 <= 18
##
## VARIABLES
## x_1 Continuous
## x_2 Continuous

prob.solve()

## 1

print("status: " + LpStatus[prob.status])

## status: Optimal

for variable in prob.variables():
     print("{}* = {}".format(variable.name, variable.varValue))


## x_1* = 2.0
## x_2* = 6.0

print(value(prob.objective))

## 36.0
```

Example: A model with unbounded solution.

```
# Example: Unbounded Optimal Solution.
# Max: Z = 3x_1 + 5x_2
# S.t.
# x_1         <= 4
# X_1 and X_2 >= 0

# Model in Python
from pulp import *
# Define the Model
prob = LpProblem("Profit", LpMaximize)
x1 = LpVariable('x_1', lowBound = 0)
x2 = LpVariable('x_2', lowBound = 0)
# Objective function
prob += 3*x1 + 5*x2, "Obj"
```

```
# Constraints
prob += x1 <= 4
print(prob)
```

```
## Profit:
## MAXIMIZE
## 3*x_1 + 5*x_2 + 0
## SUBJECT TO
## _C1: x_1 <= 4
##
## VARIABLES
## x_1 Continuous
## x_2 Continuous
```

```
prob.solve()
```

```
## -2
```

```
print("status: " + LpStatus[prob.status])
```

```
## status: Unbounded
```

```
for variable in prob.variables():
    print("{}* = {}".format(variable.name, variable.varValue))
```

```
## x_1* = 0.0
## x_2* = 0.0
```

```
print(value(prob.objective))
```

```
## 0.0
```

Example: An infeasible model

```
# Example: Infeasible Model.
# Max: Z = 6x_1 + 4x_2
# S.t.
#  x_1         >= 4
#        2x_2 >= 20
# 3x_1 + 2X_2 <= 18
# X_1 and X_2 >= 0

# Model in Python
from pulp import *
# Define the Model
prob = LpProblem("Profit", LpMaximize)
x1 = LpVariable('x_1', lowBound = 0)
x2 = LpVariable('x_2', lowBound = 0)
# Objective function
prob += 6*x1 + 4*x2, "Obj"
# Constraints
```

```python
prob += x1 >= 4
prob += 2*x2 >= 20
prob += 3*x1 + 2*x2 <= 18
print(prob)
```

```
## Profit:
## MAXIMIZE
## 6*x_1 + 4*x_2 + 0
## SUBJECT TO
## _C1: x_1 >= 4
##
## _C2: 2 x_2 >= 20
##
## _C3: 3 x_1 + 2 x_2 <= 18
##
## VARIABLES
## x_1 Continuous
## x_2 Continuous
```

```python
prob.solve()
```

```
## -1
```

```python
print("status: " + LpStatus[prob.status])

#if LpStatus[prob.status] == "Infeasible":
#    sys.exit("There is no feasible solution.")
```

```
## status: Infeasible
```

```python
for variable in prob.variables():
     print("{}* = {}".format(variable.name, variable.varValue))
```

```
## x_1* = 0.0
## x_2* = 9.0
```

```python
print(value(prob.objective))
```

```
## 36.0
```

```
# Example: WYNDOR GLASS CO.
# Max: Z = 3x_1 + 5x_2
# S.t.
#  x_1        <= 4
#       2x_2 <= 12
# 3x_1 + 2X_2 <= 18
# X_1 and X_2 >= 0

# Model in Python
from pulp import *
# Define the Model
prob = LpProblem("WYNDOR GLASS CO.",
LpMaximize)
x1 = LpVariable('x_1', lowBound = 0)
x2 = LpVariable('x_2', lowBound = 0)
# Objective function
prob += 3*x1 + 5*x2, "Obj"
# Constraints
prob += x1 <= 4
prob += 2*x2 <= 12
prob += 3*x1 + 2*x2 <= 18
print(prob)

prob.solve()
print("status: " + LpStatus[prob.status])

for variable in prob.variables():
    print("{}* = {}".format(variable.name,
variable.varValue))

print(value(prob.objective))


# **Conclusions**

# The OR team used this approach to find that the
optimal solution is $x_1 = 2$, $x_2 = 6$, with $Z = 36$.
# This solution indicates that the Wyndor Glass Co.
should produce products 1 and 2 at the rate of 2 batches
# per week and 6 batches per week, respectively, with a
resulting total profit of $36,000 per week.
# No other mix of the two products would be so profitable
*according to the model*.
```