HW4 ETM 540

Risto Rushford

2021-02-13

Exercise 4.1: Sensitivity Analysis of Adding New Drones to the Production Mix

Background: Starting Model

We are given the task of adding to the existing production mix model a new product: the Eel drone model which can be used as an aquatic submersible. Additionally, our business is adding a new finishing department, which also adds a new cost for each unit produced. The new mix is portrayed in the table below:

```
# fig.cap='A Simple DEA Model for Store Management'
Figure <- DrawIOdiagram (c("Employees\n(FTE)" ),
                         c("Sales\n($/week)"),
                         '"\nStore\n "')
tmp<-capture.output(rsvg_png(charToRaw(export_svg(Figure)),
                             'images/DEA_IO_stores.PNG'))
knitr::include_graphics("images/DEA_IO_stores.PNG")
```

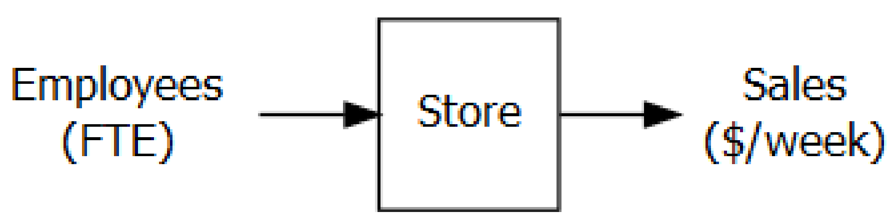

Table 1: Revised Production Plan

Characteristic	Ants	Bats	Cats	Eels	Available
Profit	$7	$12	$5	$16	
Machining	1	4	2	4	800
Assembly	3	6	2	8	900
Testing	2	2	1	25	480
Sensor	2	10	2	16	1200
Finishing	1	1	1	12	500

Model Description

Given this new breakdown of product profit and element costs, we can revise the explicit model developed in previous exercises to the following:

Max $7 \cdot Ants + 12 \cdot Bats + 5 \cdot Cats + 16 \cdot Eels$
s.t.:

$1 \cdot Ants + 4 \cdot Bats + 2 \cdot Cats + 4 \cdot Eels \leq 800$ (machining hours)
$3 \cdot Ants + 6 \cdot Bats + 2 \cdot Cats + 8 \cdot Eels \leq 900$ (assembly hours)
$2 \cdot Ants + 2 \cdot Bats + 1 \cdot Cats + 25 \cdot Eels \leq 480$ (testing hours)
$2 \cdot Ants + 10 \cdot Bats + 2 \cdot Cats + 16 \cdot Eels \leq 1200$ (sensors)
$2 \cdot Ants + 10 \cdot Bats + 2 \cdot Cats + 16 \cdot Eels \leq 500$ (finishing)
$Ants, Bats, Cats, Eels \geq 0$

Formulation 1: Explicit model of revised production variables

Building in the OMPR Engine

With this initial formulation completed, we can now enter our model into the OMPR solver engine:

```
Eels_model <- MIPModel() %>%    # Initialized empty model
  add_variable(Ants, type = "continuous", lb = 0) %>%
  add_variable(Bats, type = "continuous", lb = 0) %>%
  add_variable(Cats, type = "continuous", lb = 0) %>%
  add_variable(Eels, type = "continuous", lb = 0) %>%

  set_objective(7*Ants + 12*Bats + 5*Cats + 16*Eels, "max") %>%

  add_constraint(1*Ants + 4*Bats + 2*Cats + 4*Eels <= 800) %>% # machine hours
  add_constraint(3*Ants + 6*Bats + 2*Cats + 8*Eels <= 900) %>% # assembly hours
  add_constraint(2*Ants + 2*Bats + 1*Cats + 25*Eels <= 480) %>% # testing hours
  add_constraint(2*Ants + 10*Bats + 2*Cats + 16*Eels <= 1200) %>%  # sensors
  add_constraint(1*Ants + 1*Bats + 1*Cats + 12*Eels <= 500) # finishing

resultE <- solve_model(Eels_model, with_ROI(solver = "glpk")) # placeholder

print(solver_status(resultE))
```

```
## [1] "optimal"
```

```
Eels_result <- cbind(objective_value(resultE),
                 get_solution(resultE, Ants),
                 get_solution(resultE, Bats),
                 get_solution(resultE, Cats),
                 get_solution(resultE, Eels))
colnames(Eels_result) <- list("Profit", "Ants", "Bats", "Cats", "Eels")
rownames(Eels_result) <- list("Solution")

knitr::kable(Eels_result,booktabs = T,
             caption = 'Updated Profit with New Production Mix Elements') %>%
  kable_styling(latex_options = c("striped", "hold_position"))
```

Table 2: Updated Profit with New Production Mix Elements

	Profit	Ants	Bats	Cats	Eels
Solution	2225	50	0	375	0

Discussion of Preliminary Results

Here we see that the solver has determined that the Bat and Eel drone models are not profitable, but that by producing 50 Ants and 375 Cats we could produce a profit of $2225.

This is a similar outcome as in the Chapter 2 exercise where we added the Dog drone model, which was also determined by the solver not to be profitable for production. This raises the question, why? This calls for a sensitivity analysis.

Shadow Prices

To understand the actual value of the resources being utilized for drone production, we need to look at the **shadow prices**. This will help us to understand how best we might tweak our production model to realize a more profitable outcome.

In following with chapter 4 of the text, let's examine the row duals, aka shadow prices:

```
shadow_price <-as.matrix(get_row_duals(resultE))
dimnames(shadow_price) <- list(c("Machining",
                                 "Assembly",
                                 "Testing",
                                 "Sensors",
                                 "Finishing"),
                               c("Row Duals"))

knitr::kable(shadow_price, booktabs = T,
             caption = 'Shadow Price of Constrained Resources') %>%
  kable_styling(latex_options = c("striped", "hold_position"))
```

Table 3: Shadow Price of Constrained Resources

	Row Duals
Machining	0.25
Assembly	2.25
Testing	0.00
Sensors	0.00
Finishing	0.00

Here we see that the shadow price for testing, sensors and finishing is 0, meaning that the marginal value of one additional unit of each of these production elements is $0. This makes sense since we see that for the items being produced, none of these three elements are being maxed out. That is to say that these are not the limiting factors of the current optimal result returned by the solver.

On the other hand, we see that the cost of machining is at $0.25 per hour and the cost of assembly is at $2.25 per hour.

We see that for machining, there are $1 \cdot 50 + 2 \cdot 375 = 800$ which is the maximum number of machining hours available. Similarly, but with much more impact, we see that for assembly hours there are $3 \cdot 50 + 2 \cdot 375 = 900$, which again maxes out the total available hours. However we see that assembly hours bear a much heavier shadow price. This being the case, we will focus our analysis on this element of the model.

Adding an Hour to Assembly

As demonstrated in the text, we will experiment to find the effect of incrementing the assembly hours up by one, giving us the following revised formulation:

Max $7 \cdot Ants + 12 \cdot Bats + 5 \cdot Cats + 16 \cdot Eels$
s.t.:
$1 \cdot Ants + 4 \cdot Bats + 2 \cdot Cats + 4 \cdot Eels \leq 800$ (machining hours)
$3 \cdot Ants + 6 \cdot Bats + 2 \cdot Cats + 8 \cdot Eels \leq 901$ (assembly hours)
$2 \cdot Ants + 2 \cdot Bats + 1 \cdot Cats + 25 \cdot Eels \leq 480$ (testing hours)
$2 \cdot Ants + 10 \cdot Bats + 2 \cdot Cats + 16 \cdot Eels \leq 1200$ (sensors)
$2 \cdot Ants + 10 \cdot Bats + 2 \cdot Cats + 16 \cdot Eels \leq 500$ (finishing)
$Ants, Bats, Cats, Eels \geq 0$

Formulation 2: Explicit model with Assembly Hours incremented by 1

Then building the revised model in OMPR

```
IncrAssemHrs <- MIPModel() %>%    # Initialized empty model
  add_variable(Ants, type = "continuous", lb = 0) %>%
  add_variable(Bats, type = "continuous", lb = 0) %>%
  add_variable(Cats, type = "continuous", lb = 0) %>%
  add_variable(Eels, type = "continuous", lb = 0) %>%

  set_objective(7*Ants + 12*Bats + 5*Cats + 16*Eels, "max") %>%

  add_constraint(1*Ants + 4*Bats + 2*Cats + 4*Eels <= 800) %>% # machine hours
  add_constraint(3*Ants + 6*Bats + 2*Cats + 8*Eels <= 901) %>% # assembly hours
  add_constraint(2*Ants + 2*Bats + 1*Cats + 25*Eels <= 480) %>% # testing hours
  add_constraint(2*Ants + 10*Bats + 2*Cats + 16*Eels <= 1200) %>%  # sensors
  add_constraint(1*Ants + 1*Bats + 1*Cats + 12*Eels <= 500) # finishing

resultAssem <- solve_model(IncrAssemHrs, with_ROI(solver = "glpk")) # placeholder

print(solver_status(resultAssem))
```

```
## [1] "optimal"
```

```
IncrAssem_result <- cbind(objective_value(resultAssem),
                 get_solution(resultAssem, Ants),
                 get_solution(resultAssem, Bats),
                 get_solution(resultAssem, Cats),
                 get_solution(resultAssem, Eels))
colnames(IncrAssem_result) <- list("Profit", "Ants", "Bats", "Cats", "Eels")
rownames(IncrAssem_result) <- list("Solution")

knitr::kable(IncrAssem_result,booktabs = T,
             caption = 'Profit with Incremented Assembly Hours') %>%
  kable_styling(latex_options = c("striped", "hold_position"))
```

Table 4: Profit with Incremented Assembly Hours

	Profit	Ants	Bats	Cats	Eels
Solution	2227.25	50.5	0	374.75	0

As expected, by adding one additional hour of assembly time we have generated a new production plan which generates an additional \$2.25 of profit.

Adding an hour of Machining

Now let's look at the effect from incrementing the machining time (note that to do this we are resetting assembly hours back to 900). The linear model is:

Max $7 \cdot Ants + 12 \cdot Bats + 5 \cdot Cats + 16 \cdot Eels$
s.t.:
$1 \cdot Ants + 4 \cdot Bats + 2 \cdot Cats + 4 \cdot Eels \leq 801$ (machining hours)
$3 \cdot Ants + 6 \cdot Bats + 2 \cdot Cats + 8 \cdot Eels \leq 900$ (assembly hours)
$2 \cdot Ants + 2 \cdot Bats + 1 \cdot Cats + 25 \cdot Eels \leq 480$ (testing hours)
$2 \cdot Ants + 10 \cdot Bats + 2 \cdot Cats + 16 \cdot Eels \leq 1200$ (sensors)
$2 \cdot Ants + 10 \cdot Bats + 2 \cdot Cats + 16 \cdot Eels \leq 500$ (finishing)
$Ants, Bats, Cats, Eels \geq 0$

Formulation 3: Explicit model with Machining Hours incremented by 1

```
IncrMachHrs <- MIPModel() %>%    # Initialized empty model
  add_variable(Ants, type = "continuous", lb = 0) %>%
  add_variable(Bats, type = "continuous", lb = 0) %>%
  add_variable(Cats, type = "continuous", lb = 0) %>%
  add_variable(Eels, type = "continuous", lb = 0) %>%

  set_objective(7*Ants + 12*Bats + 5*Cats + 16*Eels, "max") %>%

  add_constraint(1*Ants + 4*Bats + 2*Cats + 4*Eels <= 801) %>% # machine hours
  add_constraint(3*Ants + 6*Bats + 2*Cats + 8*Eels <= 900) %>% # assembly hours
  add_constraint(2*Ants + 2*Bats + 1*Cats + 25*Eels <= 480) %>% # testing hours
  add_constraint(2*Ants + 10*Bats + 2*Cats + 16*Eels <= 1200) %>%  # sensors
  add_constraint(1*Ants + 1*Bats + 1*Cats + 12*Eels <= 500) # finishing

resultMach <- solve_model(IncrMachHrs, with_ROI(solver = "glpk")) # placeholder

print(solver_status(resultMach))
```

```
## [1] "optimal"
```

```
IncrMach_result <- cbind(objective_value(resultMach),
                 get_solution(resultMach, Ants),
                 get_solution(resultMach, Bats),
                 get_solution(resultMach, Cats),
                 get_solution(resultMach, Eels))
colnames(IncrMach_result) <- list("Profit", "Ants", "Bats", "Cats", "Eels")
rownames(IncrMach_result) <- list("Solution")

knitr::kable(IncrMach_result,booktabs = T,
             caption = 'Profit with Incremented Machining Hours') %>%
  kable_styling(latex_options = c("striped", "hold_position"))
```

And here we have similar results, where the profit is increased by \$0.25.

Table 5: Profit with Incremented Machining Hours

	Profit	Ants	Bats	Cats	Eels
Solution	2225.25	49.5	0	375.75	0

Adding an hour of Finishing

Now let's look at the shadow price of one of the underutilized resources by incrementing the finishing hours:

Max $7 \cdot Ants + 12 \cdot Bats + 5 \cdot Cats + 16 \cdot Eels$
s.t.:
$1 \cdot Ants + 4 \cdot Bats + 2 \cdot Cats + 4 \cdot Eels \leq 801$ (machining hours)
$3 \cdot Ants + 6 \cdot Bats + 2 \cdot Cats + 8 \cdot Eels \leq 900$ (assembly hours)
$2 \cdot Ants + 2 \cdot Bats + 1 \cdot Cats + 25 \cdot Eels \leq 480$ (testing hours)
$2 \cdot Ants + 10 \cdot Bats + 2 \cdot Cats + 16 \cdot Eels \leq 1200$ (sensors)
$2 \cdot Ants + 10 \cdot Bats + 2 \cdot Cats + 16 \cdot Eels \leq 501$ (finishing)
$Ants, Bats, Cats, Eels \geq 0$

Formulation 4: Explicit model with Finishing Hours incremented by 1

```
IncrFinHrs <- MIPModel() %>%    # Initialized empty model
  add_variable(Ants, type = "continuous", lb = 0) %>%
  add_variable(Bats, type = "continuous", lb = 0) %>%
  add_variable(Cats, type = "continuous", lb = 0) %>%
  add_variable(Eels, type = "continuous", lb = 0) %>%

  set_objective(7*Ants + 12*Bats + 5*Cats + 16*Eels, "max") %>%

  add_constraint(1*Ants + 4*Bats + 2*Cats + 4*Eels <= 800) %>% # machine hours
  add_constraint(3*Ants + 6*Bats + 2*Cats + 8*Eels <= 900) %>% # assembly hours
  add_constraint(2*Ants + 2*Bats + 1*Cats + 25*Eels <= 480) %>% # testing hours
  add_constraint(2*Ants + 10*Bats + 2*Cats + 16*Eels <= 1200) %>%  # sensors
  add_constraint(1*Ants + 1*Bats + 1*Cats + 12*Eels <= 501) # finishing

resultFin <- solve_model(IncrFinHrs, with_ROI(solver = "glpk")) # placeholder

print(solver_status(resultFin))
```

```
## [1] "optimal"
```

```
IncrFin_result <- cbind(objective_value(resultFin),
                 get_solution(resultFin, Ants),
                 get_solution(resultFin, Bats),
                 get_solution(resultFin, Cats),
                 get_solution(resultFin, Eels))
colnames(IncrFin_result) <- list("Profit", "Ants", "Bats", "Cats", "Eels")
rownames(IncrFin_result) <- list("Solution")

knitr::kable(IncrFin_result,booktabs = T,
             caption = 'Profit with Incremented Finishing Hours') %>%
  kable_styling(latex_options = c("striped", "hold_position"))
```

However this change has no effect on the profit, confirming that the shadow price of this element is not a concern.

Table 6: Profit with Incremented Finishing Hours

	Profit	Ants	Bats	Cats	Eels
Solution	2225	50	0	375	0

Reduced Cost of Variables

Now we'll examine the *reduced costs* of variables, which the text describes as "the per unit marginal profit... minus the per unit value (in terms of shadow prices) of the resources used by a unit in production." We'll examine the reduced cost by extracting the column duals from the model in OMPR:

```
reduced_cost <-as.matrix(get_column_duals(resultE))
dimnames(reduced_cost) <- list(c("Ants",
                                 "Bats",
                                 "Cats",
                                 "Eels"),
                               c("Columns Duals"))

knitr::kable(reduced_cost, booktabs = T,
             caption = 'Reduced Cost of Variables') %>%
  kable_styling(latex_options = c("striped", "hold_position"))
```

Table 7: Reduced Cost of Variables

	Columns Duals
Ants	0.0
Bats	-2.5
Cats	0.0
Eels	-3.0

Recall that the reduced cost is the difference between the value of the resources consumed by the product and the value of the product in the objective function. Recalling that all of our variables have a lower bound of zero, the Ants and Cats have zero reduced cost, while Bats and Eels each have negative reduced costs.

Reduced Cost of an Ant

Let's compare the reduced costs to the shadow prices in the context of the production of a single Ant:

```
ant_res_used<-cbind(shadow_price,c(1,3,2,2,1))
colnames(ant_res_used)<-c("Row Duals", "Resources Used")

knitr::kable(ant_res_used, booktabs = T,
             caption = 'Resources Used by an Ant and Shadow Prices of Resources') %>%
  kable_styling(latex_options = c("striped", "hold_position"))
```

Using this output we can calculate the marginal value as

$$1 \cdot \$0.25 + 3 \cdot \$2.25 + 2 \cdot 0 + 2 \cdot 0 + 1 \cdot 0 = \$7$$

Thus the profit per Ant is $7, putting the profit per Ant in balance with the marginal value, explaining why the reduced cost (difference) is zero. In other words, the value of the resources used in producing an Ant is equal to the profit earned from selling that Ant in the optimal solution.

Table 8: Resources Used by an Ant and Shadow Prices of Resources

	Row Duals	Resources Used
Machining	0.25	1
Assembly	2.25	3
Testing	0.00	2
Sensors	0.00	2
Finishing	0.00	1

Reduced Cost of an Eel

Now let's do the same analysis on an Eel, which shows a *shadow price* of $0 and a *reduced cost* −$3:

```
eel_res_used<-cbind(shadow_price,c(4,8,25,16,12))
colnames(eel_res_used)<-c("Row Duals", "Resources Used")

knitr::kable(eel_res_used, booktabs = T,
             caption = 'Resources Used by an Eel and Shadow Prices of Resources') %>%
  kable_styling(latex_options = c("striped", "hold_position"))
```

Table 9: Resources Used by an Eel and Shadow Prices of Resources

	Row Duals	Resources Used
Machining	0.25	4
Assembly	2.25	8
Testing	0.00	25
Sensors	0.00	16
Finishing	0.00	12

This situation is much more interesting. The marginal value is calculated as

$4 \cdot \$0.25 + 8 \cdot \$2.25 + 25 \cdot \$0 + 16 \cdot \$0 + 12 \cdot \$0 = \$19$

The marginal value of Eels is $3 more than the profit value, which explains the imbalance where the reduced cost is evaluated at −$3 as extracted from the OMPR model above.

Forced Eel Production

Now we can test the results of the shadow price and reduced cost analysis by reformulating our model and setting the lower bound of production for Eels to be 1. The formulation is:

Max $7 \cdot Ants + 12 \cdot Bats + 5 \cdot Cats + 16 \cdot Eels$
s.t.:
$1 \cdot Ants + 4 \cdot Bats + 2 \cdot Cats + 4 \cdot Eels \leq 801$ (machining hours)
$3 \cdot Ants + 6 \cdot Bats + 2 \cdot Cats + 8 \cdot Eels \leq 900$ (assembly hours)
$2 \cdot Ants + 2 \cdot Bats + 1 \cdot Cats + 25 \cdot Eels \leq 480$ (testing hours)
$2 \cdot Ants + 10 \cdot Bats + 2 \cdot Cats + 16 \cdot Eels \leq 1200$ (sensors)
$2 \cdot Ants + 10 \cdot Bats + 2 \cdot Cats + 16 \cdot Eels \leq 500$ (finishing)
$Ants, Bats, Cats \geq 0$
$Eels \geq 1$

Formulation 5: Forced Eel Production

Which entered into the OMPR engine is

```
EelInc1 <- MIPModel() %>%    # Initialized empty model
  add_variable(Ants, type = "continuous", lb = 0) %>%
  add_variable(Bats, type = "continuous", lb = 0) %>%
  add_variable(Cats, type = "continuous", lb = 0) %>%
  add_variable(Eels, type = "continuous", lb = 1) %>%

  set_objective(7*Ants + 12*Bats + 5*Cats + 16*Eels, "max") %>%

  add_constraint(1*Ants + 4*Bats + 2*Cats + 4*Eels <= 800) %>% # machine hours
  add_constraint(3*Ants + 6*Bats + 2*Cats + 8*Eels <= 900) %>% # assembly hours
  add_constraint(2*Ants + 2*Bats + 1*Cats + 25*Eels <= 480) %>% # testing hours
  add_constraint(2*Ants + 10*Bats + 2*Cats + 16*Eels <= 1200) %>%  # sensors
  add_constraint(1*Ants + 1*Bats + 1*Cats + 12*Eels <= 500) # finishing

resultEel1 <- solve_model(EelInc1, with_ROI(solver = "glpk")) # placeholder

print(solver_status(resultEel1))
```

```
## [1] "optimal"
```

```
Eel1_result <- cbind(objective_value(resultEel1),
                  get_solution(resultEel1, Ants),
                  get_solution(resultEel1, Bats),
                  get_solution(resultEel1, Cats),
                  get_solution(resultEel1, Eels))
colnames(Eel1_result) <- list("Profit", "Ants", "Bats", "Cats", "Eels")
rownames(Eel1_result) <- list("Solution")

knitr::kable(Eel1_result,booktabs = T,
             caption = 'Model with Forced Eel Production') %>%
  kable_styling(latex_options = c("striped", "hold_position"))
```

Table 10: Model with Forced Eel Production

	Profit	Ants	Bats	Cats	Eels
Solution	2197	38	10	359	1

And then comparing the results of the forced Eel production plan to the original case gives us

```
rownames(Eels_result) <- "Base Case"
rownames(Eel1_result) <- "Force one Bat"
temp1 <-rbind(Eels_result,Eel1_result)

knitr::kable(temp1,booktabs = T,
             caption = 'Base Case Vs Forced Eel Production') %>%
  kable_styling(latex_options = c("striped", "hold_position"))
```

Notice that the difference between profits in the two models is $\$2225 - \$2197 = \$28$. This reduction in profits occurred because forcing the one Eel reduced the resources available to make 12 Ants and 16 Cats, and replacing the amount of profit that those would have added with the profit of one Eel which is in total $28

Table 11: Base Case Vs Forced Eel Production

	Profit	Ants	Bats	Cats	Eels
Base Case	2225	50	0	375	0
Force one Bat	2197	38	10	359	1

less than before. Interestingly, it also seems to have enabled to allocation of resources to produce 10 Bats, which previously were not shown to be profitable.

Modified Objective Function

Now we can test the results of the shadow price and reduced cost analysis by reformulating our model in a few different ways. First let's try adjusting the objective function to see how that affects our results. Let's try setting the profit of Eels at \$20, thus better balancing the unit profit with the unit cost. The revised formulation is

Max $7 \cdot Ants + 12 \cdot Bats + 5 \cdot Cats + 20 \cdot Eels$
s.t.:
$1 \cdot Ants + 4 \cdot Bats + 2 \cdot Cats + 4 \cdot Eels \leq 801$ (machining hours)
$3 \cdot Ants + 6 \cdot Bats + 2 \cdot Cats + 8 \cdot Eels \leq 900$ (assembly hours)
$2 \cdot Ants + 2 \cdot Bats + 1 \cdot Cats + 25 \cdot Eels \leq 480$ (testing hours)
$2 \cdot Ants + 10 \cdot Bats + 2 \cdot Cats + 16 \cdot Eels \leq 1200$ (sensors)
$2 \cdot Ants + 10 \cdot Bats + 2 \cdot Cats + 16 \cdot Eels \leq 500$ (finishing)
$Ants, Bats, Cats, Eels \geq 0$

Formulation 6: Forced Eel Production

```
EelBal <- MIPModel() %>%    # Initialized empty model
  add_variable(Ants, type = "continuous", lb = 0) %>%
  add_variable(Bats, type = "continuous", lb = 0) %>%
  add_variable(Cats, type = "continuous", lb = 0) %>%
  add_variable(Eels, type = "continuous", lb = 0) %>%

  set_objective(7*Ants + 12*Bats + 5*Cats + 20*Eels, "max") %>%

  add_constraint(1*Ants + 4*Bats + 2*Cats + 4*Eels <= 800) %>% # machine hours
  add_constraint(3*Ants + 6*Bats + 2*Cats + 8*Eels <= 900) %>% # assembly hours
  add_constraint(2*Ants + 2*Bats + 1*Cats + 25*Eels <= 480) %>% # testing hours
  add_constraint(2*Ants + 10*Bats + 2*Cats + 16*Eels <= 1200) %>%  # sensors
  add_constraint(1*Ants + 1*Bats + 1*Cats + 12*Eels <= 500) # finishing

resultBal <- solve_model(EelBal, with_ROI(solver = "glpk")) # placeholder

print(solver_status(resultBal))
```

```
## [1] "optimal"
```

```
EelBal_result <- cbind(objective_value(resultBal),
                  get_solution(resultBal, Ants),
                  get_solution(resultBal, Bats),
                  get_solution(resultBal, Cats),
                  get_solution(resultBal, Eels))
colnames(EelBal_result) <- list("Profit", "Ants", "Bats", "Cats", "Eels")
rownames(EelBal_result) <- list("Solution")
```

```r
knitr::kable(EelBal_result,booktabs = T,
             caption = 'Model with Balanced Eel Cost') %>%
  kable_styling(latex_options = c("striped", "hold_position"))
```

Table 12: Model with Balanced Eel Cost

	Profit	Ants	Bats	Cats	Eels
Solution	2225.25	49.5	0	374.75	0.25

And then comparing the results of the forced Eel production plan to the original case gives us

```r
rownames(Eels_result) <- "Base Case"
rownames(EelBal_result) <- "Eel Balanced Cost"
temp2 <-rbind(Eels_result,EelBal_result)

knitr::kable(temp2,booktabs = T,
             caption = 'Base Case Vs Higher Eel Profit') %>%
  kable_styling(latex_options = c("striped", "hold_position"))
```

Table 13: Base Case Vs Higher Eel Profit

	Profit	Ants	Bats	Cats	Eels
Base Case	2225.00	50.0	0	375.00	0.00
Eel Balanced Cost	2225.25	49.5	0	374.75	0.25

Notice that the difference between profits in the two models is now $0.25, slightly in favor of the Eel model now, although arguably not enough to justify adding it to the product line. It may simply be that the technology and/or cost of materials (other resources not accounted for in our model) going into the production of these are too unavailable. We see this with products in the market all the time, that eventually become cheaper as technology matures or access to materials improves.

Modified Resource Usage

As our final analysis, let's look at what happens if technology actually does improve and as a result the Eel drone model becomes cheaper to produce. The machining standards have been well established and easier to accomplish, they are quicker to assemble and require less testing due to a more mature design, and the finishing cost is reduced thanks to improvements in material properties of the paints used to protect against corrosion. Let the revised formulation be

$$
\begin{aligned}
&\text{Max } 7 \cdot Ants + 12 \cdot Bats + 5 \cdot Cats + 16 \cdot Eels \\
&\text{s.t.:} \\
&1 \cdot Ants + 4 \cdot Bats + 2 \cdot Cats + 3 \cdot Eels \leq 800 \text{ (machining hours)} \\
&3 \cdot Ants + 6 \cdot Bats + 2 \cdot Cats + 6 \cdot Eels \leq 900 \text{ (assembly hours)} \\
&2 \cdot Ants + 2 \cdot Bats + 1 \cdot Cats + 10 \cdot Eels \leq 480 \text{ (testing hours)} \\
&2 \cdot Ants + 10 \cdot Bats + 2 \cdot Cats + 16 \cdot Eels \leq 1200 \text{ (sensors)} \\
&2 \cdot Ants + 10 \cdot Bats + 2 \cdot Cats + 10 \cdot Eels \leq 500 \text{ (finishing)} \\
&Ants, Bats, Cats, Eels \geq 0
\end{aligned}
$$

Formulation 7: Explicit model with finishing hours incremented by 1

```r
Eels_Red <- MIPModel() %>%    # Initialized empty model
  add_variable(Ants, type = "continuous", lb = 0) %>%
  add_variable(Bats, type = "continuous", lb = 0) %>%
  add_variable(Cats, type = "continuous", lb = 0) %>%
  add_variable(Eels, type = "continuous", lb = 0) %>%

  set_objective(7*Ants + 12*Bats + 5*Cats + 16*Eels, "max") %>%

  add_constraint(1*Ants + 4*Bats + 2*Cats + 3*Eels <= 800) %>% # machine hours
  add_constraint(3*Ants + 6*Bats + 2*Cats + 6*Eels <= 900) %>% # assembly hours
  add_constraint(2*Ants + 2*Bats + 1*Cats + 10*Eels <= 480) %>% # testing hours
  add_constraint(2*Ants + 10*Bats + 2*Cats + 16*Eels <= 1200) %>%  # sensors
  add_constraint(1*Ants + 1*Bats + 1*Cats + 10*Eels <= 500) # finishing

resultRed <- solve_model(Eels_Red, with_ROI(solver = "glpk")) # placeholder

print(solver_status(resultRed))
```

```
## [1] "optimal"
```

```r
EelRed_result <- cbind(objective_value(resultRed),
                 get_solution(resultRed, Ants),
                 get_solution(resultRed, Bats),
                 get_solution(resultRed, Cats),
                 get_solution(resultRed, Eels))
colnames(EelRed_result) <- list("Profit", "Ants", "Bats", "Cats", "Eels")
rownames(EelRed_result) <- list("Solution")

knitr::kable(EelRed_result,booktabs = T,
             caption = 'Model with Reduced Eel Resource Cost') %>%
  kable_styling(latex_options = c("striped", "hold_position"))
```

And then comparing the results of the forced Eel production plan to the original case gives us

```r
rownames(Eels_result) <- "Base Case"
rownames(EelRed_result) <- "Eels Reduced Resources"
temp3 <-rbind(Eels_result,EelRed_result)
```

Table 14: Model with Reduced Eel Resource Cost

	Profit	Ants	Bats	Cats	Eels
Solution	2226.4	48.8	0	374.4	0.8

```
knitr::kable(temp3,booktabs = T,
             caption = 'Base Case Vs Eels Reduced Resource Utilization') %>%
  kable_styling(latex_options = c("striped", "hold_position"))
```

Table 15: Base Case Vs Eels Reduced Resource Utilization

	Profit	Ants	Bats	Cats	Eels
Base Case	2225.0	50.0	0	375.0	0.0
Eels Reduced Resources	2226.4	48.8	0	374.4	0.8

With this final modification we see that by maturing the technology and methods used to manufacture the Eels we see the still incremental but better return. This seems to provide support to the logic of putting funding in research & development with the focus of technology maturation.

Comparison of Results

Finally, let's examine the results of the Sensitivity Analysis

```
rownames(Eels_result) <- "Base Case"
rownames(IncrAssem_result) <- "Increased Assembly Hours"
rownames(Eel1_result) <- "Force One Eel"
rownames(EelBal_result) <- "Eels Balanced Cost"
rownames(EelRed_result) <- "Eels Reduced Resources"
temp4 <-rbind(Eels_result, IncrAssem_result, Eel1_result, EelBal_result, EelRed_result)

knitr::kable(temp4,booktabs = T,
             caption = 'Sensitivity Analysis Aggregated Results') %>%
  kable_styling(latex_options = c("striped", "hold_position"))
```

Table 16: Sensitivity Analysis Aggregated Results

	Profit	Ants	Bats	Cats	Eels
Base Case	2225.00	50.0	0	375.00	0.00
Increased Assembly Hours	2227.25	50.5	0	374.75	0.00
Force One Eel	2197.00	38.0	10	359.00	1.00
Eels Balanced Cost	2225.25	49.5	0	374.75	0.25
Eels Reduced Resources	2226.40	48.8	0	374.40	0.80

Looking at the aggregated results after the various modification in the *Sensitivity Analysis*, we find that the best way to eek out incremental improvements is to find ways to allocate additional of constrained resources, such as Assembly Hours in the table above.

We also see that forcing Eel production results in a decrease in profit even though the profit per unit is higher than for the drone models. This is because the cost of the resources going into production of that one Eel is greater than the profit made, and additionally the forced reallocation of resources has resulted in a small number of Bat drones creeping into the optimized result. This could actually be useful information if it turns out that we do need to make a limited number of Eels, perhaps by special order from a valued customer, that some of the cost may be recouped by manufacturing a limited number of Bat drones simultaneously.

Next we consider raising the price of the Eel drones per unit to be $1 greater than the manufacturing cost. This makes the Eel drone just profitable enough that the solver now suggests making 1/4 of one unit. While this would not alone be sufficient to make use plan our production line around, it does raise the consideration that the Eel drones are undervalued.

Finally, we examine the effects of reducing the resources allocated to the Eels, in this case through technology maturation which has resulted in cheaper manufacturing processes. This has similar effect as the previous step. It's important to note that the adjustments made to the formulation above were arbitrary, however the results are still worth consideration as these suggest that reducing the manufacturing cost of the Eel drones may lead to future profitability.

The conclusion that this would likely lead me to (supposing no further information was available) is that for the present it is best to continue manufacturing according to the base case with 50 Ants and 375 Cats. However in the near term it will be worth it to invest in increasing the resource capacity of the constrained resources of Assembly and Machining hours. And finally in the long term, we can invest in R&D to improve the technology and reduce the costs of the manufacturing processes that go into the Eel drone, while researching the market to determine what features may be added or improved to increase its utility and thus fetch a higher price among consumers.