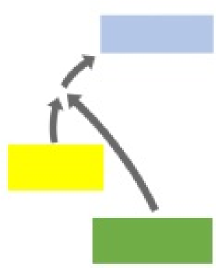
SHA575: Optimization and Modeling Simultaneous Decisions

What you'll do

- **Create an optimization model**
- **Solve a linear model in Excel using the Solver add-in**
- **Make tactical decisions using shadow prices**
- **Set prices in a non-linear model**
- **Approximate an optimal solution for a non-linear model**
- **Use summary statistics to approximate optimal results for a stochastic problem**



Course Description

Uncoordinated decision-making can be the bane of an organization. How many times have you heard of a perfectly reasonable-even data-based-decision in one part of an operation leading to a serious negative impact elsewhere?

Most businesses now recognize the value of a reliable business model that incorporates decisions that impact key business objectives. In practice, the complexity of an authentic model sometimes dictates that multiple decisions are made at the same time, and the outcomes of these decisions can interact in ways that can't easily be teased apart. In this environment, professionals need a robust, quantifiable understanding of these ripple effects in order to maximize business performance by making the best possible set of decisions.

In this course you will work with cash flow models in Excel that link objectives, multiple decisions, and resource constraints to create a realistic reflection of the decision-making environment. You will use a data analysis tool called Solver to find optimal solutions based on these models. You will use sensitivity reports to fine-tune decisions and make pricing determinations. And you will adjust models to incorporate emerging complexity and accommodate new information relevant to your model.

Chris Anderson
Professor, School of Hotel Administration
SC Johnson College of Business, Cornell University

Chris Anderson is a professor at the Cornell School of Hotel Administration. Prior to his appointment in 2006, he was on the faculty at the Ivey Business School in London, Ontario, Canada. His main research focus is on revenue management and service pricing. He actively works with industry, across numerous industry types, in the application and development of revenue management. He has worked with a variety of hotels, airlines, rental car and tour companies as well as many consumer packaged goods and financial services firms. Anderson's research has been funded by numerous governmental agencies and industry partners, and he serves on the editorial board of the *Journal of Revenue and Pricing Management* and is the regional editor for the *International Journal of Revenue Management*. At the School of Hotel Administration, he teaches courses in revenue management and service operations management.

Author Welcome

In practice, we quite often face very difficult, complex decision-making settings. And I suppose given enough time, you could eventually find a good solution to the problem. This course is going to focus on coming up with efficient, optimal solutions to those situations. We're going to focus on settings where we need to make two or more decisions simultaneously. We need to make them simultaneously because there's something that links those decisions together, whether it's limited time or constrained resources. We're going to develop your ability to recognize when we need to make decisions simultaneously versus one at a time. We're going to tackle how to efficiently model those complex settings so you can solve them quickly and then we're going to build in complexity, whether that complexity comes from uncertainty or if that complexity comes from a series of competing or multiple objectives. So I hope you enjoy the course and get excited about making simultaneous decisions.

Table of Contents

Module 1: Using Optimization

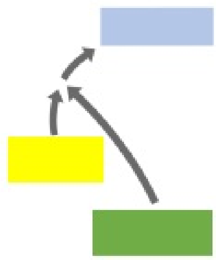
Module 2: Developing Nonlinear Models

Module 3: Creating Non-continuous Models That Work

Module 1: Using Optimization

Back to Table of Contents

Module One Introduction: Using Optimization



A cash flow model is a simulation that is a mathematical description of a business situation. Decisions are linked to objectives and to limitations set upon those decisions. Some of the model attributes are essentially fixed in your model, such as historical prices or customer order volume. Other factors, identified as constraints, have some range of allowable values. And then there are the decision variables. These are the factors for which you want to find values that lead to the best possible outcome for your objective.

Once you have built a cash flow model, you are free to 'play around' with values for the decision variables to see how the objective is impacted. But casual experimentation will rarely lead to an optimal result. And even if you do stumble on a really good solution, there's no way to be certain that you've found the true best result.

In this module, you will develop the skills necessary to move from a logical representation of your business model to a cash flow model in Excel. From there, you will use the Solver add-in for Excel to find an optimal solution for your model. In addition, you will use a sensitivity report to estimate the value of relaxing resource constraints or adding resource capacity. These calculations will equip you to determine diminishing marginal values or incremental returns on investment for your decision variables.

Back to Table of Contents

Watch: Recognizing the Need for Optimization

Optimization requires focus. Specifically, you will need to focus on a particular business objective for which you want to obtain a best numerical result. For now, we're going to focus on just a single objective to keep it simple. Often this objective is profit, or revenue, and the goal is maximize the objective. However, the objective could also be operating cost or some other value that you are trying to minimize.

Once your objective is established, you will need to consider the most important factors that influence that objective. What resource constraints exist? What are the fixed values (such as unit cost, rate of return, geographic distance) over which you have no control? And finally, what decisions will you make that might have a positive or negative effect on the objective? Often the decisions you identify will have effects that interact, so you will need to consider all these decisions at the same time.

In this video, Professor Anderson introduces a business model for a fictitious manufacturing company. The objective is to maximize profit, and the constrained resources relate to manufacturing capacity. He begins with a conceptual description of the scenario, but quickly moves into an Excel sheet to discuss specific values. For now, don't worry about how the sheet was generated or the specific values in it. If you understand the types of information he's putting in the sheet, that's enough for now. The main idea here is that constrained resources can cause decisions that might otherwise be independent to become entangled. In the next video, you'll see how the cash flow model introduced here can be used to find an optimal solution given the entangled nature of our decisions.

Video Transcript

So as humans, we're naturally predisposed to sort of do things sequentially. So if we think about making decisions, then we're quite comfortable making one decision and once that's done, moving on to our next decision. But under certain situations, we can't make decisions sort of sequentially or separately, we need to make decisions all at one time, or simultaneously. And so, let's go through an example where we illustrate when it's okay to make decisions sort of sequentially and when we need to make them simultaneously. So the example we're going to use is SpinRite manufacturing. And so, SpinRite is trying to decide how much of two different yarns it needs to make. And so, the production of yarn from nylon, is really sort of dominated by two processes, spinning and draw twisting.

And so, SpinRite has limited capacity of each of those production processes, and now is trying to decide how to allocate that production capacity in order to make those two yarns. And so, if we look at our profit from those yarns and our maximum production, then that sort of illuminates how we might go about this. So if we focus on spinning, if we take our first yarn and it has a spinning production rate in kilograms per hour, if we multiply that by the total number of spinning hours available, we'll get the maximum amount of that yarn we can spin. We can do the same for draw twisting where we take that yarns draw twisting rate, we multiply that by the total number of hours available, and so, we'll get the total number of that particular yarn that we can draw twist. And we'll do that for our second yarn. And what we noticed that for each yarn, our maximum production is limited by our spinning capacity, or our spinning hours. And so, that really means, we have this single constraining resource, spinning, which is limiting how much yarn we can make and hence, limiting how much profit we can generate. So now we have to decide how to use that spinning capacity.

So if we take each yarn's production rate, so kilograms per hour and multiply that by their corresponding contribution in dollars per hour, we basically get for each yarn how many hours, how many dollars per hour are generated through spinning. And so, we could maximize our contribution through that single constraining resource by simply choosing the yarn that had the highest dollar per hour contribution. So this is a situation where we can basically make those decisions one at a time because we have this single constraining resource and we can just maximize contribution through that resource.

Now, what happens if someone drives a forklift into one of our draw twisting machines and all of a sudden we have this substantive loss in draw twisting capacity? If we take the new draw twisting capacity in hours and multiply that by the production rate for each yarn, we end up with a new amount of yarn that we can draw twist. And what we notice, is that one of our yarns is limited by spinning i.e., the amount we can spin is less than draw twisting whereas, for our other yarn, the amount that we can draw twist is less than what we can spin. And so now, we have this situation where we have, well, one resource constraining one yarn and the other resource constraining the other yard, but both yarns need to use both resources. And so, now we have these two constraining resources and we have to decide how to use those two resources together because both our yarns need each of those resources. And so now, we're in this situation where we have to make decisions together or simultaneously versus one at a time. And we will refer to this as simultaneous decision making or as optimization.

Back to Table of Contents

Watch: Setting up and Solving Your Optimization Model

Your business model is built around an objective function whose inputs are fixed values, decision variables, and constraints that will mediate those decisions. Creating a diagrammatic representation of these inputs and how they relate to the objective can be a good way to start. The essential components of your model are in place when you have described mathematically the connections between your objective, decision variables, fixed values in your scenario, and constraints that impinge directly or indirectly on your decision variables.

You may opt to work out your model on paper first, or you may prefer to build your cash flow model directly in Excel. As you build your model in Excel, you will find that you'll benefit from clearly delineating the objective cells, the decision cells, and the constraint cells. Throughout this course, we will use a color coding scheme in which the objective cells are blue, decisions cells are yellow, and constraint cells are green. The shaded cells are all subject to change during the process of optimization.

The relationships between the factors in your model are expressed as formulas. Since the objective is dependent on the other values in the model, you will always need to place a formula in the (blue) objective cell. Similarly, (green) cells that express constrained values will always contain formulas. The (yellow) decision cells will never contain formulas, and it's a common practice is to populate all decision cells with an initial value of 1 or some other uniform value.

Once the model has been created, you'll be able to optimize your objective using Solver. In the solver pop-up dialogue window you will need to specify the cells that contain your objective, the decision variables, and the constraints. You will also need describe the nature of each constraint. Often a constraint places some upper or lower limit on some aspect of the model. The limiting values for the constraint appear in the model itself. This is done so you have the freedom to adjust them without opening the solver pop-up, and also so you can easily see what the limiting values are at all times.

In this video, Professor Anderson moves from a diagrammatic description of the model to a fully operational cash flow model in Excel. He demonstrates opening the Solver pop-up dialog box and populating it with the objective, decisions, and constraints. Finally, he runs Solver and shows an optimized result.

You may wish to download the Excel workbook referenced in this video and refer to it after watching the video. It is often helpful to look more carefully at the formulas and verify that you understand how each works, or make modifications and see how the results change.

Video Transcript

So simultaneous decision making or optimization is quite cumbersome to do manually or by hand. We really need computers to help us do that. In order to do it with computers, we need to sort of structure our decision in a certain fashion. So let's look at SpinRite manufacturing to illustrate that structure. So what do we need to decide in SpinRite? Well we need to decide how many kilograms of each of two yarns that we're going to produce. How do we know if we've done a good job? Well the best job would be the one that maximizes profit. Do we have any limits on that profit? Well yes, we only have certain available spinning and draw twisting capacity or time. And so we can now sort of take that structure and lay it out in a spreadsheet to help us sort of visualize and link those parts together.

So we can sort of set up the number of kilograms of each of our two yarns that we're going to produce. And we know that each of these yarns has a contribution in dollars per kilogram. So the product of kilograms times dollars per kilogram is our profit for each yarn. We can sum that up for our total profit and we want to make that as large as possible. Now we're constrained or limited by our spinning and draw twisting time. When we can calculate how much spinning time we've used, by taking the kilograms for each yarn and dividing that by the production rate or kilograms per hour, for each of those yarns to get the total number of hours of spinning time used to produce each yarn, we can sum that up to get our total amount of spinning time used, and we know that that has to be less than or equal to the spinning time that's available. And we can do that for draw twisting as well, looking at how much, how many draw-twisting hours we've required to produce each of these kilograms of yarn across both types. And so basically now we have our cash flow or our spreadsheet that links our two decisions to our profit as well as to our constraints. And if we wanted to, we could just play around with our spreadsheet and go through the numerous combinations of different values of yarn we could produce, keeping track of our contribution, while making sure we don't violate this spinning and draw-twisting capacity available.

Instead of that we're going to use Solver, which is an Excel add in that will help us solve this computationally. So when we open up Solver, we have this window that's sort of laid out in the same logic as our spreadsheet. We have our profit and we want to maximize that. We're going to maximize that by changing our two decisions, how many kilograms of each yarn we're going to produce? And then we have these constraints. We can add in these constraints, where the spinning time we used has to be less than or equal to the spinning time that's available, the draw twisting time that we use has to be less than or equal to the draw twisting time that's available. And because we're doing this computationally, we also have to tell our computer that, hey the smallest amount of yarn we can produce is zero, right. So each of our yarns has to be, our yarn productions has to be greater than or equal to zero. And so we can simply click on solve. And then our computer will go through and now determine the profit maximizing set of yarn quantities to produce in order to make sure that we haven't used more spinning or draw twisting time, more than is currently available.

Note: Depending on the instance of Excel on your computer, Solver may or may not have been added to the toolbar.

See the Using Solver Excel step-by-step for full instructions on how to access and use Solver in Excel. Versions of Excel older than '97 do not support Solver. Third party solvers, such as OpenSolver, are also available and may have slightly different performance characteristics.

Back to Table of Contents

Activity: Find an Optimal Result Using Solver

Now that you've seen how Solver can be used to find an optimal value for an objective function, take a few minutes to practice setting up and running Solver on an already-completed cash flow model. This is ungraded activity, and you will not need to turn in any work to your instructor—but if you run into trouble completing this activity, you should reach out to your instructor for guidance.

In this model you will see that there are four manufacturing Plants (1–4), each of which uses a particular component in their manufacturing process. If it helps to have a specific example in mind, imagine each Plant makes tail light assemblies for a different model of car, but they all use the same type of lightbulb in their assemblies. There are five Suppliers (A–E) that can supply the bulbs, and all suppliers' bulbs are interchangeable. The five Suppliers have five different MANUFACTURING COSTS for the bulbs. In addition, each Supplier passes on a different unit SHIPPING COST depending on which of your Plants it supplies with bulbs.

Your objective is minimize TOTAL COSTS while meeting the monthly UNITS REQUIRED constraint determined by each Plant. There is an additional set of constraints: each Supplier has a limited number of UNITS AVAILABLE to ship each month. Follow the steps below to set up and run Solver on this model.

1. Download the Excel workbook.
2. Navigate to the Data tab and select the Solver button. If you cannot find the Solver button, review the *Installing Solver* section of the Using Solver Excel step-by-step instructions.
3. In the Solver dialog box, select the field labeled Set Objective, enter the value **$C$20**. This sets the blue TOTAL COSTS field as your objective.
4. Below the Set Objective field in the Solver dialog, make sure the radio button next to **Min** is selected, because you want to minimize costs.

5. Define the decision variables in the field labeled "By Changing Variable Cells:" by entering **$C$2:$G$5**.
6. Select the **Add** button to add your first set of constraints. A second dialog will appear. In the left field, enter the range **$H$2:$H$5**. In the middle field, select ./mo /math'>. In the right field, select **$I$2:$I$5**. Now select **Add** to add the constraint to Solver. This assures that the UNITS SHIPPED values in each row are at least as large as the UNITS REQUIRED values, so each Plant gets its required supply of bulbs.
7. Because you selected the Add button, you are still in the add constraints dialog and ready to add your other constraint. In the left field enter **$C$6:$G$6**. In the center field enter ./mo /math'> if it is not already showing. In the right field enter **$C$7:$G$7**. Select **OK** to add the constraint and return to the main Solver menu. This second constraint assures that the model doesn't count on getting more SUPPLIED UNITS from each Supplier than they have UNITS AVAILABLE.
8. Make sure the box labeled **Make Unconstrained Variables Non-Negative** has a checkmark in it. This prevents the model from inserting negative values (of bulbs) in the model.
9. Select **Solve** to run Solver. If you get an optimized Objective value of $32,285, then celebrate. You have successfully used Solver. If you got another value or an error, use Save As to store a copy of this attempt and start again from step 1.

Back to Table of Contents

Course Project, Part One—Create a Linear Optimization Model

You've seen how Solver can be used to find an optimal solution to a linear model, provided the model is correctly built. In this first part of the course project, you will build a model and specify Solver parameters for an optimized solution. Your instructor will review your model for correctness so you can successfully complete additional project parts later in the course. *Completion of all parts of this project is a course requirement.*

Instructions:

1. Download the course project document and the Excel workbook template.
2. Complete Part One of the course project.
3. Save your work.
4. Submit your partially completed project document and the workbook with your model for grading and credit.

Do not hesitate to contact your instructor if you have any questions about the project. You will add to this document as the course proceeds and will submit it to the course instructor at the end of the course.

Before you begin:

Before starting your work, please review the **rubric** (a list of evaluative criteria) for this assignment. Also review eCornell's policy regarding **plagiarism** (the presentation of someone else's work as your own without source credit).

Back to Table of Contents

Watch: Examining Constraints Using Shadow Prices

When you prepare a model to support optimization of the objective function, you are incorporating a set of limits on your decisions in the form of constraints. The constraints may not all be written in such a way that they limit a specific decision variable to a given range. Collectively, though, you can think of the constraints like a malleable fence that encloses an interior whose area you are trying to maximize or minimize. There are situations in which you would like to move a fencepost—that is, to increase (or decrease) a constrained variable to improve the outcome of your objective function—but your constraint will not allow this change. Constraints that limit your current set of possible decisions are known as binding constraints.

In this video, Professor Anderson discusses binding constraints and how they relate to the optimization problem. He demonstrates how to generate and interpret a sensitivity report in Solver that exposes the binding constraints in an optimization model. Within the sensitivity report, he focuses specifically on a set of values, called shadow prices, that expose the incremental impact of modifying constraint values in the model.

You may wish to download the Excel workbook referenced in this video and refer to it after watching the video. It is often helpful to look more carefully at the formulas and verify that you understand how each works (**in particular, notice the use of the =SUMPRODUCT( ) function in the objective cell**), or make modifications and see how the results change.

Video Transcript

So when we have to make two or more decisions, and those decisions are linked by a set of constraints, then we need to use our optimization framework. One of the outcomes of our optimization framework is the ability to quantify the impact of those constraints upon our objective, the thing we're trying to maximize or minimize. So let's illustrate that with Koana Coffee Company. So Koana is basically trying to decide how to use a set quantity of beans. Now, these beans are of two qualities— high and low quality. And then Koana is going to take this set of beans and use those to produce three blended forms of coffee: premium, natural and house blend.

Now, each of these three blends has slightly different contributions, where the contribution is highest for premium, a little bit lower for natural and the lowest for house. Now, there's some limitations, some quality limitations on the number of low and high quality beans, which goes into our premium and natural roast, whereas house blend, we can use any sort of combination of beans, and we also have finite demand for each of these three products. So a sequential decision maker might simply decide, "Well, let's make as much of the premium roast coffee as possible because it has the highest margin." Whereas our simultaneous decision maker would realize, "No, I need to decide how much of each of these three products to make together because these three products use a constrained resource of beans. They have limitations on how we mix those beans together, and there's only a limited market for each of these three products." And so we can structure that in our sort of optimization framework in our spreadsheet. So we actually have six decisions: how many low and high quality beans are going into each of our three products? We know we've made the right decisions when we've maximized our contribution.

So taking those volumes produced multiplied by their contributions per kilogram. And we have some limitations upon our contribution. Those limitations are the available demand for each of our three blends, as well as our available supply of the two types of beans and then how we mix them for our premium and natural. We can go and open up Solver, ask Solver to maximize our contribution by changing our decisions, given our set of constraints. We hit solve, and Solver comes back and gives us our profit maximizing solution, which is cool in itself. Now, one of the things we notice is that some of these constraints are what we call binding. In that, if we look at the supply of high and low quality beans, we've used all those beans. And so those beans are limiting our contribution. So we call those binding constraints. We can look at the impact of those constraints upon our contribution by relaxing those a little bit. So let's add another kilogram of high quality beans to our beans that are available, rerun Solver, and look at the difference in our contribution. That difference is the incremental or marginal impact of that constraint upon our contribution.

And we could do that for all our binding constraints. We could go through each of these constraints that is limiting our contribution and see what is the incremental impact upon our contribution of that constraint. Now, fortunately, we don't have to do that sort of manually. Our optimization framework will do that for us. So after we've executed Solver and asked it to solve, it comes back with a window displaying our results. And one of the reports we can generate is called the sensitivity report. If we asked to generate that report, what comes back is this table of these incremental values. In the optimization world, we call these shadow prices. And, here, we see that incremental contribution of having more high quality beans. What we notice is there's that incremental contribution of high quality beans is, in fact, more than the highest contribution of any of the three products we're selling. So it will be hard for us ahead of time to realize that, but because of the interplay of the constraints, we get this incremental impact, which is higher than any of those contributions.

Back to Table of Contents

Watch: Making Tactical Decisions with Shadow Prices

Assuming your model is structured in a way that allows it, running Solver allows you to generate a sensitivity report. Within that report you'll find a table labeled Constraints, and it's here that you'll find information on shadow prices. But if you've already found the optimal solution for your model, what good is all this extra information?

The key to understanding the value of shadow prices is realizing that your constraints are not always going to be firmly fixed values. So if you could obtain more of a resource, or if you have a competing offer for a component of your product, what would it be worth to you to deviate from your original plan? Exploring the boundaries of your model allows you to consider ways to further improve on your optimal solution. This kind of boundary exploration is a technique included in what's known as sensitivity analysis or post-optimality analysis.

In this video, Professor Anderson describes an airline ticketing scenario in which shadow prices allow the value of flights into and out of a hub to be considered individually, allowing the airline operator to make real-time decisions about seat pricing. To understand the benefit of using shadow prices in this example, in can be helpful to think of the optimization model as the static approach (often the status quo) and post-optimality analysis as a dynamic approach that relaxes some of the assumptions of the original model.

You may wish to download the Excel workbook referenced in this video and refer to it after watching the video. It is often helpful to look more carefully at the formulas and verify that you understand how each works, or make modifications and see how the results change.

Video Transcript

So in optimization, we refer to the incremental impact of constraints upon our objective, the thing we're trying to maximize or minimize, as shadow prices. And shadow prices add a lot of insight into the incremental impact of those constraints. But under many circumstances, we can actually use those shadow prices themselves to make decisions. So let's use Ontario Airlines as an example. So Ontario operates a small network and they have a series of planes. Each of these planes has a set number of seats. And planes fly, for example, from Phoenix to Houston and then that plane turns around and goes back from Houston to Phoenix. We have two other planes which are flying from Chicago to Houston and from Miami to Houston. And so now in addition to those simple little legs, consumers could also fly from Phoenix to Chicago but they would have to connect in Houston. And so our airline has a set of prices across those different itineraries and it has forecasted demand across all those itineraries.

So it's forecasted how many people who want to go from Chicago to Miami at set prices. But the airline has to decide, well how many seats should be available across each of these itineraries. And they have to sort of use optimization to do that because these decisions are linked, because anybody who wants to fly from Chicago to Miami basically could be sitting beside somebody who wants to fly from Chicago to Houston or beside somebody who wants to fly from Chicago to Phoenix, right. So those three decisions are related because those people are all potentially on the same plane. So we're going to use Solver to maximize revenue by deciding how many seats to allocate on each plane to these different itineraries by making sure that we don't exceed the plane capacities and at the same time that what we allocate is less than or equal to demand. So we can actually use Solver to get those allocations. But one of the cool things about optimization is it will also return these incremental impacts of those constrained resources. And so in this setting we have six constrained resources. We basically have the capacity on each of those six legs that the airline is flying. And we'll notice for Phoenix to Houston that shadow price or incremental value is in fact zero because we have fewer consumers on that leg than we have available capacity. And it turns out that we have 42 surplus seats. And so the airline could sell those 42 seats for any value and it would still be profitable. But if we look at some of our other legs, we notice that they have non-zero shadow prices, because we've used all those available resources.

Now those shadow prices tell us the marginal impact of a seat on each of those individual legs. And so if a consumer wanted to fly from Chicago to Miami, well they would consume a piece of capacity on the flight from

Chicago to Houston and in consuming that capacity they would cost us that shadow price or that marginal value. They would also consume a piece of capacity from Houston to Miami and so they would cost us that marginal value of shadow price from Houston to Miami. And so the airline would not want to set prices or sell seats that were lower than the sum of those marginal values for that itinerary for that consumer flying from Chicago to Miami. Once we know the shadow prices, we actually know the lowest available prices that the airline would ever consider posting for different itineraries. So we actually don't need to know the allocations. We don't need to know how many seats we're selling across these itineraries. We just need to know these marginal values. And so now we know what are the lowest prices that our airline would consider posting.

Back to Table of Contents

Read: Interpreting the Sensitivity Report's Constraints Table

- **Allowable increases and decreases set bounds on how much a constraint can change without affecting the shadow price.**
- **Non-zero values in the shadow price column indicate binding constraints.**
- **Careful choice of headers can make reading the Constraints table easier.**

You've seen how shadow prices can be used to enhance the decision-making value of your model. Now let's take a more comprehensive look at the Constraints table so you are familiar with some of the other information it conveys.

30	Constraints						
31			Final	Shadow	Constraint	Allowable	Allowable
32	Cell	Name	Value	Price	R.H. Side	Increase	Decrease
33	$C$6	SUPPLIED UNITS Supplier A	18500	-0.22	18500	1000	18500
34	$D$6	SUPPLIED UNITS Supplier B	15000	-0.16	15000	1000	4500
35	$E$6	SUPPLIED UNITS Supplier C	1000	0	26500	1E+30	25500
36	$F$6	SUPPLIED UNITS Supplier D	27000	-0.25	27000	1000	4500
37	$G$6	SUPPLIED UNITS Supplier E	18500	-0.15	18500	500	4500
38	$H$2	Plant 1 UNITS SHIPPED	19000	0.63	19000	4500	500
39	$H$3	Plant 2 UNITS SHIPPED	15000	0.61	15000	4500	1000
40	$H$4	Plant 3 UNITS SHIPPED	22000	0.58	22000	4500	1000
41	$H$5	Plant 4 UNITS SHIPPED	24000	0.6	24000	25500	1000
42							

The Constraints table shown here is from a sensitivity report generated for the auto lightbulb example used in the activity you completed earlier. Notice that each constraint from the problem appears in its own row. Because the rows were carefully labeled, it's relatively easy to understand from the Name column what is being referred to within each constraint row. When you work in the Constraints table, you generally will be considering values in only one row at a time.

The **Final Value** column shows the value reached for each constrained variable when the model's objective has been optimized. Looking at row 39, you can see that Plant 2 was shipped 15000 units. This is equal to the value in the **R.H. Side** (right hand side) of the constraint. R.H. side, or RHS refers to the value that is set as the limiting value.

Non-zero values in the **Shadow Price** column indicate binding constraints. The left hand side (LHS) and RHS values are the same, and the constraint prevents further increase (or decrease) of that variable's value. All but one of the constraints in this example are binding. As you've seen, the shadow price represents the cost (or benefit) of incremental increasing (or decreasing) the limiting value, or "Right Hand" side of the constraint. But this shadow price does not hold for arbitrarily large changes in constraint value. The range over which shadow prices are valid is specified in the table's **Allowable Increase** and **Allowable Decrease** columns. Look back at row 39 in the Constraints table above. Allowable Increase and Decrease values indicate that if demand dropped by 1000 or increased by 4500, each unit change within that range would result in a change in the objective function by 0.61. Because we are minimizing costs, if we decreased the RHS by 1 our costs would decrease by 0.61.

Consider what would happen if Plant 2 suddenly needed just 1500 bulbs a month-a tenth of the 15000 bulbs it currently requires. Since the required number of units has decreased by 13500, which is 15000 -1500, and 13500 > 1000, the first 1000 bulbs that are not needed are going to decrease total costs by 1000*.061 = $61. But we don't know the impact of the remaining 12500, which lie outside the allowable range. We could determine this impact is by re-running solver with the RHS set to 1500. This would give us a new objective value. The difference between the old and new objective values divided by 13500 is the average reduction per unit. This value will be less than 0.61. The sensitivity report would then tell us the new incremental value of the 1500th unit.

Back to Table of Contents

Watch: Gaining Insight from the Variable Cells Table

A sensitivity report includes two tables: Variable Cells and Constraints. While the Constraints table exposes shadow prices, the Variable Cells table allows you to focus more directly on each of the decision variables. Specifically, you gain insight into the contribution of each variable (which is usually multiplied by one or more fixed values and moderated by one or more constraints) toward your objective. This aspect of sensitivity analysis opens up many avenues to explore "what if" questions about the decision parameters in your model.

In this video, Professor Anderson explores a scenario that provides an optimal result for a complex trans-shipping problem. But the result, while optimal given all factors and constraints built into the model, recommends at least one decision that the company considers infeasible for reasons that the model was not designed to predict. Rather than reformulate the model (with significantly more complexity), Professor Anderson shows how to tinker with the results to take into consideration the problematic result.

This example exposes an aspect of optimization that is important to understand. Namely, that running Solver is not necessarily the endpoint of the optimization process. After running Solver there is interpretation, both in the form of sensitivity analysis and in coming to terms with business implications of the decision variables. After this, there can be a modification of the model's structure, or there can be additional tinkering that leaves the structure intact but modifies individual cells within the structure.

You may wish to download the Excel workbook referenced in this video and refer to it after watching the video. It is often helpful to look more carefully at the formulas and verify that you understand how each works, or make modifications and see how the results change.

Video Transcript

So optimization tells us the best decisions in order to maximize or minimize our objective function. Sensitivity analysis in optimization can tell us the incremental impact of constraints upon that objective function. It can also tell us the impact of the parameters of our model upon the decisions we make. And so let's illustrate that through an example. So Abitibi-Price is a large manufacturer of newsprint. It produces newsprint at three different plants, Buffalo, Tampa and Fort Worth, and then it ships that newsprint from those three plants to eight distribution centers. Now, the three different plants have different manufacturing costs, and there's differing transportation costs from each plant to each of the distribution centers. And so, Abitibi needs to meet demand but it wants to do so in a efficient or cost-minimizing way. And so they have to make 24 decisions, how much to ship from each mill to each distribution center. They have to make sure that they don't ship more from a mill than they have newsprint available or production capacity. And at the same time, they have to make sure that what they ship to each distribution center at least meets demand. And they're going to do that while minimizing total costs.

And so, they're naturally in this optimization framework. And so, they run optimization or on our case they run Solver. And what they find is, is that basically newsprint produced at Tampa is only shipped to one location. And so they're thinking that perhaps they want to sort of diversify some of the production at Tampa. And so, they might ask, under what conditions would we change our decisions? Under what conditions would we ship more newsprint from Tampa? So if we were to run Solver and ask for our sensitivity analysis, we would get the impact of those decisions upon our objective function or the impact of those variable cells upon our objective function. And what we see here are a set of sort of threshold values. So initially, if we focus say on one of those distribution centers.

Let's look at Louisville and Tampa. And we'll notice that the total cost to service Louisville from Tampa is the sum of the production cost at Tampa, plus the shipping costs from Louisville to Tampa. And what our sensitivity analysis provides us is a threshold, such that if that cost was to decrease by that threshold, then we would change our decision. Right now our decision is zero. So if we go back to our model and we decrease the shipping costs by that threshold and rerun Solver, we notice now that we will ship from Tampa to Louisville. So we've changed that decision given we've changed that relationship between the decision and the objective function. So we can get the impact of all those different combinations of shipping and production costs upon the 24 decisions that we have to make through our sensitivity analysis.

Back to Table of Contents

Read: Interpreting the Sensitivity Report's Variable Cells Table

- **Allowable increases and decreases set bounds on how much an objective coefficient can change without affecting the value of the decision variable.**

- **The contribution of a decision variable to the objective is measured by taking the product of the objective coefficient and the final value.**

You've seen how values in the Reduced Cost column of the Variable Cells table can be used to find a threshold that will allow a decision to change. Now we're going to consider how some of the other information in the table helps complete the picture. As with Constraints tables, you will be generally reading across a row to get a more complete understanding of what the table's values mean.

	Cell	Name	Final Value	Reduced Cost	Objective Coefficient	Allowable Increase	Allowable Decrease
9	$C$2	Plant 1 Supplier A	0	0.02	0.43	1E+30	0.02
10	$D$2	Plant 1 Supplier B	500	0	0.47	0.01	0.02
11	$E$2	Plant 1 Supplier C	0	0.03	0.66	1E+30	0.03
12	$F$2	Plant 1 Supplier D	0	0.01	0.39	1E+30	0.01
13	$G$2	Plant 1 Supplier E	18500	0	0.48	0.02	1E+30
14	$C$3	Plant 2 Supplier A	0	0	0.39	1E+30	0
15	$D$3	Plant 2 Supplier B	10000	0	0.45	0	0.01
16	$E$3	Plant 2 Supplier C	0	0.06	0.67	1E+30	0.06
17	$F$3	Plant 2 Supplier D	5000	0	0.36	0.01	0.01
18	$G$3	Plant 2 Supplier E	0	0.03	0.49	1E+30	0.03
19	$C$4	Plant 3 Supplier A	0	0.01	0.37	1E+30	0.01
20	$D$4	Plant 3 Supplier B	0	0.03	0.45	1E+30	0.03
21	$E$4	Plant 3 Supplier C	0	0.02	0.6	1E+30	0.02
22	$F$4	Plant 3 Supplier D	22000	0	0.33	0.01	0.58
23	$G$4	Plant 3 Supplier E	0	0.05	0.48	1E+30	0.05
24	$C$5	Plant 4 Supplier A	18500	0	0.38	0	1E+30
25	$D$5	Plant 4 Supplier B	4500	0	0.44	0.02	0
26	$E$5	Plant 4 Supplier C	1000	0	0.6	0.02	0.15
27	$F$5	Plant 4 Supplier D	0	0.04	0.39	1E+30	0.04
28	$G$5	Plant 4 Supplier E	0	0.02	0.47	1E+30	0.02

Each decision has its own row in the table. A zero value in the **Final Value** column indicates that the decision in that row made no contribution to the objective function. The **Objective Coefficient** expresses the incremental cost (or benefit) of each unit change in the Final Value of the decision variable. So the contribution to the objective is the product of the Final Value and the Objective Coefficient. Looking at row 27 of the table, Supplier D makes no shipment to Plant 4 so there is no contribution to the Objective. In row 26, though, we see that Supplier C does ship to Plant 4, and the contribution is 0.6 * 1000 = 600.

The **Allowable Increase** and **Allowable Decrease** values bound the variables in a way that is similar to their role in the Constraints table. In this case, allowable increase and decrease describe the range within which the current solution (**Final Values** for the decision variables) will not change even if the value of the objective function impact changes. Decision variables not impacting the objective (those with zeros for values) will have a non-zero **Reduced Cost**. This reduced cost tells us how much the objective function impact needs to vary before we start using that decision. In our example, starting to use a decision would mean beginning to ship from a supplier to a plant. So in row 28 we will start to ship from Supplier E to Plant 4 if we reduce the cost by at least $0.02 per unit. Similarly, in row 26, as long as the Objective Coefficient doesn't increase by 0.02 to 0.62, the decision to ship 1000 units will stay the same.

Back to Table of Contents

Interpret and Use Shadow Prices

You have seen how shadow prices can be used to make tactical decisions by relaxing constraints. In particular, you looked at a scenario involving Ontario Airlines in which you considered minimum seat prices.

In this quiz, you will check your skill with shadow price interpretation in a scenario that builds on the Ontario Airlines example. You should not need the full workbook to complete the quiz, but if you would like to download it you can reference this workbook. This is the same workbook whose link is included on the "Making Tactical Decisions" page in this course.

Read the following scenario carefully and make sure you understand it before continuing on to the quiz questions. In the questions you will be provided with a cash flow model and will be asked questions based on the model.

Scenario:

You are an operations manager for Ontario Airlines. You have been contacted by an Air France representative with a codeshare proposal. Air France has daily flights into Houston but at present is not flying to Chicago, Miami, or Phoenix. In an effort to reach these other markets, Air France is trying to negotiate an agreement with you that allows Air France to sell seats on Ontario's routes to Air France customers flying into Houston. Based on your pricing model, you will need to determine the value of Air France's offer.

You must achieve a score of 100% on this quiz to complete the course. You may take it as many times as needed to achieve that score.

Back to Table of Contents

Analyzing within Your Area of Expertise

The course project threading through this course will focus you on skills-building using a pre-defined problem and a provided data set. But it's natural (and important) to think in terms of a problem that is meaningful to you. In this discussion, take an opportunity to consider how optimization can be useful for you.

Instructions:

You are required to participate in all discussions in this course.

Discussion topic:

Select one aspect of your business or your life outside work that involves complex, interconnected factors that require simultaneous decision-making. Without going as far as making a fully-realized model, briefly describe the scenario and include in your description a best initial guess at the objective, constraints, and decision variables.

Create a post in which you:

briefly describe the decision making scenario
include a best initial guess at the objective, constraints, and decision variables you must include in the model
describe any tactical decision-making you anticipate engaging in as part of a possible post-optimality analysis

Review the posts made by other students. For at least one other student's post:

ask a question to encourage more clarity about the scenario or about one of the factors included in the model; OR
share an insight that might inform the structure of the model or suggest structural or tactical changes to the model

To participate in this discussion:

Use the **Reply** button to post a comment or reply to another comment. Please consider that this is a professional forum; courtesy and professional language and tone are expected. Before posting, please review eCornell's policy regarding **plagiarism** (the presentation of someone else's work as your own without source credit).

Back to Table of Contents

Course Project, Part Two—Perform a Sensitivity Analysis

Your examination of sensitivity reports has introduced you to using shadow prices (in the Constraints table) and reduced cost values (in the Variable Cells table) to get more utility out of your model even after finding an optimal solution. With practice, you will begin to develop instincts for how best to use this information in post-optimality analysis. In this project part, you will revisit the model you created in Part One. Your instructor will review your sensitivity analysis to verify your understanding of the model and how to use it. *Completion of all parts of this project is a course requirement.*

Instructions:

1. Open the course project document and Excel workbook you saved and submitted in Part One.
2. Complete Part Two in the project document.
3. Save your work.
4. Submit your project document and Excel workbook to your instructor for grading and credit using the Submit Assignment button on this page..

Do not hesitate to contact your instructor if you have any questions about the project. You will add to this document as the course proceeds and will submit it to the course instructor at the end of the course.

Before you begin:

Before starting your work, please review the **rubric** (a list of evaluative criteria) for this assignment. Also review eCornell's policy regarding **plagiarism** (the presentation of someone else's work as your own without source credit).

Back to Table of Contents

Module One Wrap-up: Using Optimization

In this module, you practiced describing a business scenario using a mathematical model. You identified an objective along with constraints and decisions that defined a function for that objective. You built a cash flow model of the scenario in Excel, and you set up parameters in Solver to support an optimized solution for the model. You examined sensitivity reports, interpreting shadow prices and using them to make tactical decisions.

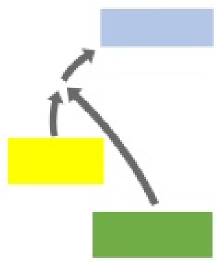
The skills you've built in this first module have already prepared you to deal with complex, interconnected decisions in an extremely efficient matter. As you become familiar with the process of setting up models and specific Solver parameters, the benefit of this method will only increase. Up to now, however, you have only dealt with linear problems—that is, with objectives whose functions are linear combinations of the decision variables. In the next module, you will begin to examine non-linear solver methods. Non-linear methods are important because many common problems cannot be solved using linear optimization.

Back to Table of Contents

Module 2: Developing Nonlinear Models

Back to Table of Contents

Module Two Introduction: Developing Nonlinear Models



As you develop a cash flow model, linearity is one of the characteristics you should strive for. As you will see, there are distinct advantages to a linear model over a nonlinear model. But it will not always be possible to build an authentic model without allowing for some nonlinearities in either the objective function or constraints. You need to be able to identify ways to avoid nonlinearity when building your model, recognize when your model is unavoidably nonlinear, and correctly apply nonlinear solution methods.

This module begins with a simple, straightforward explanation of what nonlinearity is, so don't be intimidated if the term is unfamiliar. Once you understand the basics, you'll focus on a scenario in which the addition of new constraints forces a linear model to become nonlinear. As you work through this module, understand that there are many ways that your model can become nonlinear, and that this module is only an introduction to the concept of nonlinear optimization. Finally, you'll practice applying multiple Solver methods to get the best possible optimization results from your model.

Back to Table of Contents

Watch: Recognizing Nonlinearity and its Implications

Up to this point in the course, we have dealt with models that are strictly linear. Essentially this means that our objective functions and constraints have been built by adding, not multiplying, decision variables. You might be thinking "Wait, my cash flow model used the =SUMPRODUCT( ) function, so there absolutely were products!" True, linear models do include products with decision variables, but the decision variables are multiplied by scalar (fixed number) values only. But once a function includes, say, a decision variable multiplied by itself, your model may become non-linear. Linear solution methods can no longer be used.

In this video, Professor Anderson considers what happens when a model includes an objective function that is not, or may not be, linear.

Video Transcript

Linear programming or LP for short, is a special case of optimization. In linear programming, all aspects of the model are simple linear combinations of our decisions. Our objective function and all our constraints are simply some adding and subtracting version of our decision variables. So, let's take Koana Coffee Company. So, there we had six decisions. How many low and high quality beans to go into our premium, natural, and house blend coffee? Our objective function was to maximize contribution or maximize profit. And that was simply those six decisions, times each of their contribution margins. All our constraints were also linear combinations of those six decisions. So, take the number of high quality beans used for example. That was simply the sum of high quality beans used in premium plus the high quality beans in natural plus the high quality in house. So, all aspects of our model were simple linear combinations of those six decisions. The upside of having a linear model is, when we run Solver and select our simplex LP as our solving method, we quickly get our solution and we get these insightful sensitivity analyses. But sometimes, when we run Solver, Solver comes back and tells us that the linearity conditions for our model are not met. So, that doesn't think our model was linear. Well, when might our model not be linear? Well, it might be inherently non-linear.

Say for example, we're trying to maximize revenue where, revenue is sales times price. Now, our decision variable is price but sales are a function of price. So, our objective function is going to be sales times price, where that sales is also a function of price. So, it might be a function of price squared. So, it's no longer linear in price, it's some non-linear function. Or, we might have a model which is a linear combination of those decision variables but we might have inefficiently formulated our model and included some logic, say like, if statements in Excel. They will result in our model becoming non-linear. Our goal should always be to have a linear model, if at all possible, because they're easier and faster to solve and we get these insightful post optimality sensitivity analyses around our constraints or the parameters of our model. But, if in fact our model is non-linear, that's okay. We just need to take a slightly different approach when solving it.

Back to Table of Contents

Watch: Setting Prices in a Nonlinear Model

A typical challenge when setting the price of a product is that the price has an impact on demand. The lower the price of a product, the greater the demand. This phenomenon may be expressed as price elasticity. If your revenue model expresses demand as a function of price and your objective is revenue, this results in your objective function becoming nonlinear.

In this video, Professor Anderson describes a hotel room pricing scenario in which elastic price-dependent demand leads to a nonlinear cash flow model. He demonstrates how to use one of Solver's nonlinear solution methods to obtain a solution.

You may wish to download the Excel workbook referenced in this video and refer to it after watching the video. It is often helpful to look more carefully at the formulas and verify that you understand how each works, or make modifications and see how the results change.

Note: If you run Solver on the model in the workbook linked above with the Evolutionary method select, you may see results that differ from those shown in the video. This difference can occur because the Evolutionary Solver has an element of random selection built into the algorithm. You should also be aware that with nonlinear algorithms you may need to wait more than a few seconds to see a result depending on the specifics of the model and the settings used.

Video Transcript

In optimization problems, it's always advantageous to have our objective function in our constraints, linear in our decision variables. But, quite often, or sometimes, that's not the case and either our constraints or our objective function may be non-linear combinations of our decision variables, but that doesn't mean we can't find a solution. So let's look at an example. So we have a hotel and it's looking at how to best set room prices. It has three different types of rooms: standard, junior suites, and king suites. And at sort of current prices, we have a base demand for those three different types of rooms.

At the same time, we have an estimate of their price plasticity or the percent change in demand given a percent change in price. And so given this price sensitivity of demand, our hotel is trying to see if they can do better than their current prices. Doing better would mean having higher revenue, so can we maximize revenue by changing prices? Now, one of the tricky parts here is that demand is going to be a function of price owing to that price sensitivity, and then revenue, given that it's demand times price, is now going to be a non-linear function of price, potentially price squared. And so when we formulate our optimization model by changing our prices to maximize our revenue while satisfying some constraints around price differences and available rooms across the different types, we're going to have to change the Solver approach that we're using. If we selected our simplex LP to solve this problem, we would get back a statement from Solver saying that the linearity conditions are not satisfied.

Basically, our solution approach is telling us that it can't find an answer. So we need to change that solution methodology. And so if we were to select the Evolutionary Solver, and then click on solve, then our, in this case, Solver is going to go away and look for a solution. It's going to take it a while to find that solution. Because it can't exactly find the solution, it sort of looks at many different possible solutions in an effort to find one that's better. And, in fact, when it's done its job of looking for possible answers, it comes back and Solver will tell us that it can't improve upon the current solution, all constraints are satisfied. So that means it's found a feasible solution. All the constraints are satisfied and it's returning back the best solution it could find, but it has no idea if that is the best overall solution. But given the approximations it's making, this is the best solution it was able to determine in the time that you allocated for it to look for our solution.

Back to Table of Contents

Watch: Recognizing the Limitations of Non-linear Optimization

In the hotel room pricing example, Professor Anderson used the Evolutionary Solver method to obtain a valid solution for the model. If this had been a linear model, the LP method would have delivered a clear optimal solution. When the Evolutionary method is applied to a nonlinear model, Solver often returns a solution that is valid and is a pretty good answer. But this solution may or may not be the absolute best solution possible. Given that we're trying to optimize, this 'pretty good' result is not very satisfying.

In this video, Professor Anderson looks at how the Evolutionary algorithm works and why it cannot deliver a true optimized solution every time. He contrasts Solver's Evolutionary with another nonlinear method called GRG Nonlinear. Then, he describes a method involving both nonlinear algorithms that you can use to obtain a result that is likely better than you would get using either method by itself.

Video Transcript

The evolutionary solver in Excel comes from a class of optimization solution methods referred to as metaheuristics. Different types of metaheuristics includes tabu search, neural networks and genetic algorithms. All these approaches basically generate a series of potential decisions and then based upon those values for those decision variables, they evaluate the constraints as well as your objective. And then they keep the best solutions from that set they generate that satisfy those constraints. And then based upon that subset of really good solutions, they use intelligent rules to sort of generate other candidate solutions in an effort to find the best one. As they generate more solutions, they compare the current objective value for the objective function value for that most recent set to the older ones as long as we tend to improve upon our solution we continue to look. At some stage we can't improve upon our most recent best solutions, so we stop looking for better solutions. We don't necessarily know we found the best one, but we know that given the rules that we're using we can't seem to find better ones than the one we have now.

Now there's also a different class of solution and approaches that are based upon gradients or derivatives If you think about things in the terms of calculus. And these approaches, instead of generating a whole mess of possible solutions and picking and choosing the best, they look at a potential solution and then they see how if they just sort of slightly tweak that one then in which direction does the objective function improve.

And so, you could think of these as analogous to climbing a hill. So when I'm at a specific spot on the hill, I look around and I find a direction where I'm going up the hill and so I move in that direction. I continue to move in directions that are taking me up versus down. At some point I can no longer go up, I can only go down so now I know that I've reached the top of the hill. So these gradient-based methods know when they've reached the top of the hill that they're on. Now they don't necessarily know that they've reached the top of the tallest peak, they just know that they've reached the top of the hill that they're on. So one can now combine these sort of metaheuristic and gradient-based approaches where I might use the evolutionary approach to find a really good solution to get me to the mountain that I think is the tallest, and then use the gradient-based solution to sort of go the final few steps and climb to the top of that mountain. So if we had a hotel pricing example where we're trying to set the prices across three different room types, we could formulate our optimization model. That model is going to be non-linear because revenue is a function of price and demand and demand is also a function of price. So we use our evolutionary solver to get us really close. And then, we can go back to us resolve this problem. But this time we select our gradient-based approach and then we take a final few more steps to get to the top of the mountain or our set of prices that optimize revenue.

Back to Table of Contents

Read: Sensitivity Reports with Nonlinear Methods

- **Nonlinear methods provide limited or no sensitivity reporting. No allowable increases or decreases are provided in any case.**
- **Nonlinear sensitivity report values are approximations, whereas linear report values are exact values for the optimal solution.**
- **Lagrange multipliers provide information similar to shadow prices, but they become less useful the more nonlinear results diverge from optimal values.**
- **The Evolutionary nonlinear method does not provide a sensitivity report.**

As you've seen, Solver's linear LP Simplex method provides sensitivity reports that are rich with information. In particular, shadow prices and reduced cost fields are very useful. With nonlinear methods you can also generate reports, but these reports are not nearly as useful as the linear method sensitivity report.

When you use the GRG nonlinear method and generate a sensitivity report, you'll notice a few differences right away. The sensitivity reports shown here are from the same (linear) Koana Coffee Company model. The more complete report was generated by Simplex LP. The other report is from the GRG method. The first thing you probably notice is that

the report from GRG lacks allowable increases and decreases. This means you have no way of knowing the valid range for your shadow prices, and you don't know how much your reduced costs can change without affecting the decision variable.

Variable Cells

Cell	Name	Final Value	Reduced Cost	Objective Coefficient	Allowable Increase	Allowable Decrease
$C$4	KGs_premium	333334.6667	0	3.5	2.25	0.75
$C$5	low kilograms	166667.3333	0	3.5	4.5	1.5
$C$6	high kilograms	166666.3333	0	2.5	0.75	2.25
$C$7	low kilograms	333332.6667	0	2.5	0.375	1.125
$C$8	high kilograms	0	-3	2	3	1E+30
$C$9	low kilograms	2000000	0	2	1E+30	0.5

Constraints

Cell	Name	Final Value	Shadow Price	Constraint R.H. Side	Allowable Increase	Allowable Decrease
$C$12	Premium kilograms	500002	0	3000000	1E+30	2499998
$C$13	Natural kilograms	499999	0	1000000	1E+30	500001
$C$14	House kilograms	2000000	0.5	2000000	249999.5	250000.5
$C$16	high quality kilogram:	500001	4.5	500001	499999	250001
$C$17	low quality kilograms	2500000	1.5	2500000	250000.5	249999.5
$C$19	Premium kilograms	0	-1	0	250001	499999
$C$20	Natural kilograms	0	-1	0	500001	249999.5

Simplex LP (linear) sensitivity report

Variable Cells

Cell	Name	Final Value	Reduced Gradient
$C$4	KGs_premium	333333.3333	0
$C$5	low kilograms	166666.6667	0
$C$6	high kilograms	166666.6667	0
$C$7	low kilograms	333333.3333	0
$C$8	high kilograms	0	-3.000357226
$C$9	low kilograms	2000000	0

Constraints

Cell	Name	Final Value	Lagrange Multiplier
$C$12	Premium kilograms	500000	0
$C$13	Natural kilograms	500000	0
$C$14	House kilograms	2000000	0.5
$C$16	high quality kilogram:	500000	4.499999866
$C$17	low quality kilograms	2500000	1.499999955
$C$19	Premium kilograms	0	-1
$C$20	Natural kilograms	0	-1

GRG Nonlinear sensitivity report

You can also see that the constraints table has no column labeled Shadow Prices. In its place are a set of values that are *approximately* the same as shadow prices. These values appear in a column labeled Lagrange Multiplier. What we call shadow prices are in fact a specific value of Lagrange multipliers: they are the value of the Lagrange multiplier when your solution is optimal. What this means is, if your GRG Nonlinear solution is not close to the optimal value, the Lagrange multipliers in your sensitivity report will be less useful than actual shadow prices.

In the Variable Cells table, Reduced Gradient values approximate the Reduced Cost values in a linear sensitivity report. As with Lagrange multipliers, these values become less reliable when the nonlinear solution is not close to the true optimal solution.

Evolutionary methods for nonlinear models do not provide any sensitivity reports.

Back to Table of Contents

Watch: Applying Multiple Solver Algorithms

Sometimes building an accurate cash flow model requires the use of functions that don't behave predictably throughout the range of values you need to consider. Noncontinuous functions, as they're called, are particularly problematic for the GRG Nonlinear method.

In this video, Professor Anderson discusses the challenge posed by noncontinuous and nonsmooth functions. He revisits the hotel room pricing scenario to demonstrate how poorly-behaved functions can arise. Finally, he demonstrates how using both Evolutionary and GRG Methods in sequence can allow you to work around these problematic functions.

Video Transcript

So when solving optimization problems, several solution approaches require that we have continuous functions, right, that the objective function is a continuous function of our decision variables. And further to that, some approaches also require that that objective function be a smooth function. So, a continuous function is one that has no jumps. So, if our objective function is moving and all of a sudden either has a big step up or a step down, that would be a discontinuous function. A non-smooth function would be one that say suddenly changes direction. So if Y is a function of the absolute value of X then, when X equals zero, Y suddenly changes direction. So that's a non-smooth function. We can create spreadsheet models that have these discontinuities and non-smooth functions if we start to put a lot of logic into our spreadsheets. Logic through things like Absolute Values (ABS) or MINs and MAXes, things like IF statements or we round up numbers where we use Excel functions like HLOOKUP or VLOOKUP. So all these sort of logic-based approaches can create problems for our solution methods.

So, let's look at a hotel that's trying to set optimal prices across three different types of rooms and in an effort to sort of use consumer behavior it's decided that all the prices should end in nines. And while that seems very logical, it's very hard mathematically to put that constraint into our optimization model. So, if we were doing this in our spreadsheet we could use a combination of left and right functions to determine what was the rightmost number and then put a constraint in that that number had to equal nine. Now if we run Solver and we were to select our gradient-based solution approach under numerous settings, Solver would come back and say that it can't find a feasible solution. There is no answer to your problem. Whereas if we specify the evolutionary solver, it would take some time but it would look around and come back with not only a feasible solution but a very good solution. Now that we have this very good solution, we could go back and rerun with our gradient-based solver and as long as it didn't move too far and didn't fall off a cliff it would improve upon our prior solution.

Back to Table of Contents

Activity: Execute Solver with a Nonlinear Model

As you have seen, running Solver is just as simple whether you use a linear method or a nonlinear method. The only adjustment you make is changing the selection in a dropdown menu. But getting useful results from Solver in the nonlinear case is not a trivial exercise. You need to understand how any given nonlinear method works so that you can understand its potential limitations for your specific model. Getting a good result can require experimentation and some trial and error.

In this activity, you will run Solver on the provided nonlinear model and attempt to achieve the best result possible. As you complete this activity, keep careful notes. You will be asked to share your result and the methods used to obtain them in a class discussion.

Scenario:

The scenario used for this activity involves decisions about where to build two new lab facilities to handle medical sample testing for ten hospitals. Each hospital's X and Y location is plotted on a 400 square mile grid. For the purposes of this scenario, assume travel from the hospitals to the labs will occur on grid roads spaced exactly one mile apart.

The objective is to minimize the total monthly travel needed to deliver all samples to the Lab Sites. Hospital locations and monthly trip frequency are provided. Constraints are added to require that each hospital is paired with exactly one lab for all deliveries and that each lab serves exactly five hospitals.

To complete the activity:

Step 1: Download the Nonlinear Medical Lab Siting Excel workbook

Step 2: Review the formulas in the objective and constraint cells. Open the Solver dialog box and make sure you understand how the model and optimization parameters are set up.

Step 3: Experiment with different Solver methods and techniques for combining these methods. Here are some suggested approaches to finding the best solution:

- Set all initial decision variables to 0 and record the results. Rerun solver with all decision variables set to 1 and compare the results from these two attempts.

- Try different techniques that involve alternating back and forth between Evolutionary and GRG Nonlinear methods.

- Experiment with options (for example, try using the Multistart option for the GRG Nonlinear method). Solver Tips has more information about the Options dialog.

Step 4: When you have found what you think is the best possible value for the objective, save a screenshot that includes the objective, decision, and constraint cells. Change the filename of the screenshot to "SolverBestResult_NAME.jpg" where NAME is your last name or your initials.

Refer to the Solver Tips tool and ESS How-to guide for Solver as needed. Keep in mind that the best approach to finding an optimal solution is the approach that works best. Don't forget to keep careful notes on how you achieved your results.

Back to Table of Contents

Compare Results of Nonlinear Solver Methods

Sometimes there is no substitute for practice and tinkering. With optimization problems, this is especially true when nonlinear methods are involved. You just completed an activity in which you tried different approaches to finding an optimal result. In this discussion you will share with your classmates the approach that gave you the best results.

Instructions:

You are required to participate meaningfully in all discussions in this course.

Discussion topic:

To review, the scenario in the nonlinear Solver activity involved decisions about where to build two new lab facilities to handle medical sample testing for ten hospitals. Create a post in which you:

share a screenshot of your best result (upload the "SolverBestResult_NAME.jpg" you saved earlier*
describe in as much detail as possible the approach you took (methods used, and in what sequence, as well as any changes you made using the Options dialog) to get the result

After all (or most) students have posted, review other students' results, and 'Like' any results you find that are better than your own results. This should push the best results toward the top of the discussion.

**Note: The Embed Image icon in your text editor can be used to add your screenshot to your post.*

To participate in this discussion:

Use the **Reply** button to post a comment or reply to another comment. Please consider that this is a professional forum; courtesy and professional language and tone are expected. Before posting, please review eCornell's policy regarding **plagiarism** (the presentation of someone else's work as your own without source credit).

Back to Table of Contents

Module Two Wrap-up: Developing Nonlinear Models

Many business problems require a departure from the relative simplicity of a cash flow model that is strictly linear. Where possible, you should strive for simplicity and linearity in your model. But you also need to be prepared to deal with (i.e. find optimal solutions for) nonlinear models.

This module introduced the concept of nonlinear optimization and gave you an opportunity to get some experience with nonlinear models. You ran Solver on a nonlinear model and experimented with techniques for using nonlinear methods. You also took a step backward into the realm of linear models and performed a sensitivity analysis of a linear model. Hopefully seeing the power of the sensitivity analysis had the effect of helping you appreciate some of what is given up when moving to a nonlinear model.

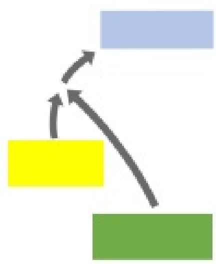
Once you are in the realm of nonlinear models, it is possible to build in arbitrarily complex restrictions. In the next module, you will encounter some approaches to making your model more authentic without adding too much complexity.

Back to Table of Contents

Module 3: Creating Non-continuous Models That Work

Back to Table of Contents

Module Introduction: Creating Noncontinuous Models That Work



In linear models, the value of the objective function changes smoothly as any given decision value changes. This predictability is what allows certainty that the optimal solution is without question the best result given the model's structure, constraints, and fixed parameters. But often the world is not predictable. Likewise, the decisions we need to make often do not have values that lie along a continuum. They are often restricted to integer values (...-1, 0, 1, 2, 3, etc.) or are *yes/no* or *go/no-go* decisions.

In this module, we will introduce noncontinuous decision variables into optimization models. This will sometimes require careful thought about how these variables can best be incorporated without rendering your model unsolvable. We will also look at ways to circumvent some of the inconvenient realities of using Solver for optimization.

Back to Table of Contents

Watch: Posing Integer Constraints

Certain resources can't be considered in fractional quantities. Manufacturing lines that require extensive retooling may operate in ways that are tracked on the scale of days, or even whole weeks or months. Dwelling units are generally not divisible into parts. And depending on the situation, individuals in the workplace often need to be considered strictly as present (one Full-Time Equivalent, or FTE) or absent (zero FTE).

In this video, Professor Anderson begins with a linear model that makes no assumptions about integer constraints for personnel. He then shows how the model needs to be adjusted to take into account the real-world situation being described. Specifically, he explores how the model can account for the reality that on a given day, an employee is either absent or present—either contributing to the total number of personnel needed, or not. The reality of the situation forces a constraint (employees present) in the model to be strictly an integer value.

You may wish to download the Excel workbook referenced in this video and refer to it after watching the video. It is often helpful to look more carefully at the formulas and verify that you understand how each works, or make modifications and see how the results change.

Video Transcript

Having linear functions definitely makes optimization easier, as we can find solutions faster and we know we have the best possible solution. Now, sometimes we will have an objective function which is a linear combination of our decision variables, but then, we introduce discontinuities through constraints that require our decision variables to be integers or whole numbers. So we have a linear solution but we have discontinuities. So, let's look at an example. So, we're trying to schedule our workforce. So, we're looking at the upcoming week, and for each day of the week we have a certain number of employees who are required to be working each of those days. Now, we're trying to figure out how many staff that we need to meet those required number of employees. But our staff, they all sort of work five days on in a row followed by two days off.

So someone who starts their shift on Monday will work through to Friday, having Saturday and Sunday off. And so, not only do we have to sort of decide how many employees we need. We also have to decide well, which day of the week do they start working. So we could lay out our model where we can determine who's working when and then how many people are working on each of the seven days. And so, now we have this set of decisions about how many people that we need to start on each of the seven days of the week. We have some constraints that the some of the people who are available has to be at least as many as we need. And then, we will have an objective that is going to minimize the number of staff required. And so, if we go and solve this, we can solve it as a linear model and we'll very quickly get a solution. But what we'll realize, that we have a lot of fractional people who are starting their shifts at different days of the week. And so, we obviously can't have fractional people, given that we're talking about full-time employees. So, one approach would be to simply go in and manually take all those fractional people and round them up to whole people.

But when we do that, we'll notice we need a couple of more employees than our original sort of partial employee sort of solution. So we can go back now to our Solver model and we can add a constraint, where you can add a constraint that all those decision variables have to be integers. And so, we can run Solver and it will find a solution that all constraints and optimally conditions are satisfied. So it will find the best possible solution under those sets of constraints. What's different about this solution is our sensitivity report is limited. We will no longer get our shadow prices for those marginal values of those constraints because they are not defined, given we've put those integer requirements on our decision variables. So we're still linear, we just have to sort of solve it a bit differently.

Back to Table of Contents

Watch: Modeling Constraints with Binary Decisions

You just saw how a binary condition (employees that are either present or absent) can force an integer value for a constraint. Often you will need to make decisions that are themselves binary, and you will want to incorporate the binary decision directly into your model. Once you see how binary values can work with decision values, it is possible to extend this idea to refine your model by adding additional decision variables.

In this video, Professor Anderson explores a scenario in which a candy manufacturer is making decisions about outsourcing product packaging. He introduces binary decision-making in two very distinct ways in this example. First, he looks at a set of competitive bids and uses binary decisions to choose among the bids offered by different suppliers. After finding an optimal result for bid selection, he adds complexity to the model by introducing a second, completely separate set of binary decisions that has the effect of managing a new constraint on the problem. Re-running Solver after adding this constraint gives a slightly less ideal result, but it carries the benefit of including a desired constraint in the model. Note that adding constraints to a model can only reduce the performance of the objective function—e.g. increase costs, reduce profit, etc.

The second point in this video can't be overemphasized. Your decisions are often driven by some specific business logic, and this logic will typically be based on binary decisions. Once you understand how to incorporate binary decisions into your model, you can continue to add logic until you reach the limits of complexity that Solver can handle.

As with other videos based on Excel workbook examples, you may wish to download the workbook and examine it more closely. For this video, we are including two slightly different versions of the same workbook because the model undergoes a substantial change in the course of the video. The initial version of the Excel workbook includes binary decision variables and Solver parameters that reflect the initial constraints on the model. The second version of the workbook includes new binary decision variables to manage the constraint of using a minimum number of suppliers used.

Video Transcript

It many optimization settings, decision variables may not be numeric in nature. They may be yes or no or accept, reject type decisions. Fortunately we can model these as special cases of integer variables, basically being binary integers, or one, zero where one is for yes, and zero is for no. So let's look an example using binary decision variables. So Mars is a large confectionery company looking at reducing its supply chain costs. So it hosts a auction, where suppliers submit bids for a set of required materials. Mars has to decide which of these bids it should accept and reject, in order to minimize its costs. So we can set up our model where we have a series of one-zero, yes-no binary decisions, and then as a function of those decisions we can determine how much of each of our inputs are acquired, basically whether or not we accepted a bid in the requirements of that bid. And so now we can simply have our optimization model, which is changing these one-zero decision variables, we have to add a constraint to our model that says these values actually have to be binary. So we have to tell Solver that they are going to be ones or zeros. And then, we put a constraint on that the sum of the products or the products that we've accepted through these bids has to be greater than or equal to our needs. So now we have a very efficient linear way of determining which bids to accept. Now Mars may also be interested in more long-term issues, where they're wanting to ensure that they have, you know, sufficient relationships with suppliers over the long term. So maybe they want to have the minimum set of suppliers from each auction that they accept bids from.

So we could look at all the suppliers that bid on this particular auction, and then count how many bids we accepted from each of those suppliers. And then we could think about trying to set up a constraint on the number of suppliers we're using. It becomes a bit tricky here though in how we translate say, I've accepted two bids from one supplier, and translating that to a one, did I accept bids from that supplier? We could do that with something like an IF statement. If the number of bids accepted from a supplier is greater than or equal to one, then a one else is zero. And then we can sum those up, and put a constraint on that in our optimization model. But having that constraint being a function of an IF statement is going to cause all sorts of problems when we try and solve this because we're basically going to have this non-smooth discontinuous function in our optimization model, which means it be very difficult to find a solution. One way around that is adding a new set of decisions, right, a set of binary, one-zero decisions for each of these suppliers. Right? So now we're going to have a new set of decision variables, they're also going to be binary. And then we're going to put some constraints on, some additional constraints on these binary decision variables that they have to be less than or equal to the number of bids accepted from each of these suppliers. Right? So if I accept zero bids from a supplier, this decision variable has to be zero, if I accepted two bids from a supplier then this decision variable would be a one.

I can sum up those decision variables. That's the number of suppliers I'm using. I could put a constraint on the sum of the number of suppliers that I've utilized. And so now we can rerun our model, our cost will be a little bit higher, but it still solves very quickly because it's still linear. Right? Those IF statements are no longer part of our our model and we simply have a new set of binary decision variables, which we've used to build in this logical base constraint. Right. So the nice part about binary decision variables is we can use them to model yes-no type decisions, and we can also use them to model logic-based constraints in our optimization framework.

Back to Table of Contents

Watch: Accommodating Multiobjective Decisions

So far we've been following a strict rule: in optimization only a single objective is permitted. In fact, a single objective is a hard requirement of all Solver algorithms, and that is not going to change.

There are, however, ways to work around this restriction. In this video, Professor Anderson describes three approaches you might take to get a single result that can be parsed into distinct values for two or more KPIs.

Video Transcript

So optimization requires a single unifying objective, you know, maximizing profit or minimizing cost. But in many situations, we may have secondary or sometimes competing objectives. So we can still use optimization in these situations, we just have to modify our approach, and so let's look at three of those modifications. So let's take an example where we have two different stakeholders whose interests are aligned but have slightly different preferences. For example, we could look at an exchange, any sort of exchange.

But let's talk about see a timeshare exchange, where owners of hotels across different locations are looking to exchange their timeshare with other timeshare holders. So I'm the operator of the exchange, I want to facilitate as many of these hotel to hotel exchanges as possible, but the actual individuals being, participating in the exchange process, they have desirabilities or different levels of desirabilities across the exchanges that could happen. They might have a set of preferences. At one level, the operator is just yes-no, a set of binary decisions, did the exchange happen? And our other stakeholder has a preference list, this is my most desirable exchange, second or third, we might assign some weightings to those, you know: one three and five. And so we could put those together where our objective now is the product of those ones and zeros and those preferences. And we would maximize the sum of those products. And so now we have a single unifying objective which is going to maximize the number of exchanges but at the same time, ensure that those exchanges are incorporating the preference of our other stakeholders. In other situations, the objectives of our stakeholders may not be necessarily aligned, or we may, ourselves, have differing objectives. So take a traditional finance application where we're trying to decide how to invest our retirement into a series of different investment stocks or bonds or whatever they might be.

And so at one level, we want to maximize our return, but at another level, we want to minimize our risk. We want to make sure we don't lose all our money. Those are very distinct competing objectives. So one way to tackle this problem is I'm going to still put together a optimization model where I decide which investments to participate in. I'm going to minimize my risk, but then I'm going to put a constraint on my return. So minimize my risk given that I have at least a 10% return. So we put one of our objectives in as a constraint and then we've maximized or minimized our other objective. So here we again have these competing objectives which we can accommodate into our optimization framework. Now in other settings, we could take those two different competing objectives and kind of put them on the same playing field, and we could come up with a weighted average or a combination of those two objectives. And our goal now would be to maximize that combination of those individual objectives. So let's look at a non-optimization setting to put it in context. So you're looking at trying to choose which flight for an upcoming trip. You could either choose flights which were really inexpensive, or maybe flights that were really short.

Now, we need to sort of manage these two conflicting objectives, time and money. We could put those together into one sort of efficient flight, where we now translate time into some sort of monetary equivalent with some kind of factor. And so now, we're going to minimize this new objective, which is some combination of the price and some combination of the time. And so we have to sort of put time and money on the same playing field, and then we could minimize this new objective, which is the combination of those. We would do that in our regular optimization setting, where now we have this single objective which accommodates both sets of these conflicting objectives.

Back to Table of Contents

Watch: Using Summary Statistics to Approximate Optimization in Stochastic Problems

Earlier in this module, we considered how to add complexity to an existing model. Some problems, however, are inherently too complex for Solver if modeled with their native precision. The standard version of Solver that is provided as an Excel add-in can handle up to 200 decision variables. And while there is a fee-based version of Solver that can handle more decision variables (at present as many as 8000), even this seemingly large number of variables can be surpassed by certain kinds of scenarios.

In this video, Professor Anderson discusses a class of problems known as stochastic optimization problems. Stochastic models have randomness built into the model parameters in place of the fixed scalar values we've considered in this course. This randomness can generate very complex models for which Solver is unable to find a solution. Optimal solutions for stochastic problems is a topic beyond the scope of this course and is in fact an active area of research. Rather than trying to solve stochastic optimization models, Professor Anderson uses a specific example to illustrate how to use approximation to side-step the stochastic nature of a problem and build a solvable model that will give reasonable results.

Back to Table of Contents

Watch: Avoiding Solver Limitations with Using Standard Form

When you're considering a complex model and approximation is not desirable, you may need to look beyond the capabilities of Solver. One of the advantages of Solver is that the visual nature of the spreadsheet makes model development easier for beginners. Another advantage is the ready availability of Excel. But more capable, commercial optimization tools do exist. One thing to be aware of is that in order to use commercial optimization tools efficiently, you will need to structure your model using mathematical constructs called matrices.

In this video, Professor Anderson discusses alternatives to Solver and illustrates how to structure your model in standard form using matrices, a universally used mathematical representation for optimization modeling.

Video Transcript

So in our simultaneous decision making approach, we need to be able to link our decisions to the objective and the constraints. And so typically, we have this one-to-one mapping, given a decision what is the resulting revenue and what are their constraints associated with that? Now in many settings, we don't necessarily know the exact outcome of a given decision or choice. There might be a range or a set of possible outcomes. So those outcomes are uncertain. So we have this stochastic set of outcomes given a decision. In general, we refer to this optimization framework, where we have uncertain outcomes as stochastic optimization. And so while stochastic optimization is a ongoing research area, we can take stochastic problems and distill those into our traditional deterministic framework, or ones where we know the attributes and solve those.

So if we look at a very simple example where we're trying to set prices for two products, and let's make it even simpler and say that each of those products only has three possible prices because we like prices to say end in nines. And then for each of those three possible prices, there's a set of possible different sales we might realize. Let's just assume there's only

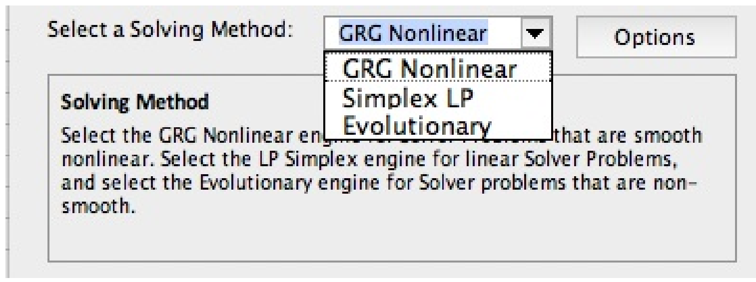
10 possible levels of demand at each of those three prices for each of those two products. And so even though we have the simplest of examples, with just two products and three prices, we quickly see that there's a lot of different outcomes that we have to look over. And so in stochastic optimization, we're trying to maximize the expected profit by looking at the profit that results and weighting it by each of the probabilities associated with each of these outcomes. And so while that might be doable in our simple example, as our examples get more realistic in size, then this quickly becomes intractable.

And so one way to modify this problem, is to take our stochastic problem and translate it to a deterministic one, where we basically take these 10 possible demand outcomes and translate those to the average sales for a given price. And so now, we've just dropped this complexity considerably. So instead of searching over all possible demand levels and their associated probabilities, we just compact that into one average demand at each of the three prices for each of our two products. And so all of a sudden, now we have a very simple optimization problem that we can solve rather quickly. So this is one way to address a stochastic problem, basically, making it deterministic and allowing us to solve it quickly.

Back to Table of Contents

Tool: Non-linear Solver Options

Solver has a series of parameter settings that help improve how it performs optimization. These settings are invoked by selecting Options in the Solver input window.



In Excel Solver, there is an Option dialog box that allows custom settings to be used during the solving process. You can set options across all methods or specifically for one of the two nonlinear solution approaches. In the tabs below, you can see options that apply to each nonlinear method (GRG and Evolutionary) as well as some settings that can be applied to both methods.

This page outlines a few useful pointers that will help you navigate Solver's options. You can download the associated Solver Tips tool for access to the same information on the go. For detailed background and support on all options for Solver see the Support Pages created by Frontline Systems, the developers of Excel Solver.

All Methods
GRG Non-linear
Evolutionary

Back to Table of Contents

Course Project, Part Three—Adjust Your Model

It's not uncommon to need to modify an optimization model to improve its authenticity or respond to new information or requirements. In this part of the course project, you will once again revisit the model you created in Part One. This time, you will adjust the model itself to incorporate more stringent requirements. *Completion of all parts of this project is a course requirement.*

Instructions:

1. Open the course project document and Excel workbook you saved and submitted previously in Part Two.
2. Complete Part Three in the project document.
3. Save your work.
4. Once you've finished, review the entire document, making any final additions or revisions, and then **submit both the project document and the Excel workbook for instructor review using the Submit Assignment button on this page.**

Do not hesitate to contact your instructor if you have any questions about the project.

Before you begin:

Before starting your work, please review the **rubric** (a list of evaluative criteria) for this assignment. Also review eCornell's policy regarding **plagiarism** (the presentation of someone else's work as your own without source credit).

Back to Table of Contents

Module Wrap-up: Creating Noncontinuous Models That Work

After beginning with the most simple examples of optimization problems, you have traveled in this course to the opposite extreme to consider problems that challenge the boundaries of what is known by the current state of research. If typical, your work in optimization will lie mostly in the middle of this spectrum. And if you're lucky, you'll be able to address most of your optimization questions using linear models. If not, you've had an introduction to at least a few topics in nonlinear optimization.

In this module, you extended an existing optimization model using binary decision variables to build in a new constraint. This incremental step in complexity reflects the nature of the optimization modeling process. To the extent that you can make modest, low-cost improvements to your model, you will continue to do so *provided your model continues to produce usable results*. As you continue in this area of work, you will hopefully continue to explore what is possible with optimization and what techniques and tools might serve you well.

Back to Table of Contents

Read: Thank You and Farewell

Chris Anderson
Professor
School of Hotel Administration
SC Johnson College of Business, Cornell University

Congratulations on completing *Optimization and Modeling Simultaneous Decisions*. Hopefully you found this course both useful and enlightening. Optimization is a deep topic. While this course barely scratches the surface of what is possible, I hope you'll agree that even modest effort with optimization modeling can offer great rewards in terms of efficiency and improved decision-making. Best wishes as you continue along your learning path.

From all of us at Cornell University and eCornell, thank you for participating in this course.

Sincerely,

Chris Anderson

Back to Table of Contents