*DS 775*

# Prescriptive Analytics

## Sensitivity Analysis

## The Need for Sensitivity Analysis

- Postoptimality Analysis
- Sensitivity Analysis
- Sensitive Parameters

Recall

**Certainty Assumption:** Model parameters are known constants

Section 4.7 introduces the idea of postoptimality analysis, investigating the final model to see if it can be improved. This includes performing sensitivity analysis. Sensitivity analysis is necessary, because the certainty assumption of linear programming is rarely ever satisfied.

Recall that the certainty assumption says that the model parameters are known constants. This is just not realistic. The parameters that are assumed to be known are more likely to be estimates found through descriptive and/or predictive analytics.

Sensitivity analysis is done to identify sensitive parameters. Sensitive parameters are the ones that would change the optimal solution if they are different from what was first thought. This means that, if the original estimates are off, the optimal solution that is found initially may not actually be the true optimal solution.

# Sensitivity for Each Type of Parameter

$$\text{minimize} \quad Z = c_1x_1 + c_2x_2 + \cdots + C_nX_n$$

$$\text{subject to} \quad \begin{aligned} a_{11}x_{11} + a_{12}x_{12} + \cdots + a_{1n}x_{1n} &\le b_1 \\ a_{21}x_{21} + a_{22}x_{22} + \cdots + a_{2n}x_{2n} &\le b_2 \\ &\vdots \\ a_{m1}x_{m1} + a_{m2}x_{m2} + \cdots + a_{mn}x_{mn} &\le b_m \end{aligned}$$

$$\text{and all } x_{ij} \ge 0.$$

"Binding" means it holds with equality at optimal solution

Hopefully, you recall that in the standard form of a linear programming problem, there are three different kinds of parameters. The a sub ij's are the coefficients of the constraints. The b sub i values represent the constraint boundaries. These are also called the right hand side values and often represent the amount of a resource available.

The c sub j's are the coefficients in the objective function itself. The IOR tutorial gives a nice visual demonstration for the sensitivity of the c sub j's when there are two design variables in the model. This will be shown in a later example. It's a nice idea to get in your head, because the general idea is carried over into models with more variables but cannot be displayed graphically.

The a sub ij parameters can be checked for sensitivity by seeing if the constraint is binding at the optimal solution. A binding constraint means that the a sub ij's in that constraint are sensitive. Non-binding constraints indicate that the a sub ij's in that constraint are not sensitive.

Oftentimes, changing the a sub ij's is negligible. Whereas, changes to the b sub i's or c sub j's when they are sensitive parameters can have a higher impact on the response variable. In the Excel sensitivity report, the top table is for the c sub j's and the bottom table is for the a sub ij's and b sub i's. Sensitivity analysis often begins with an investigation of the b sub i values.

## Shadow Prices

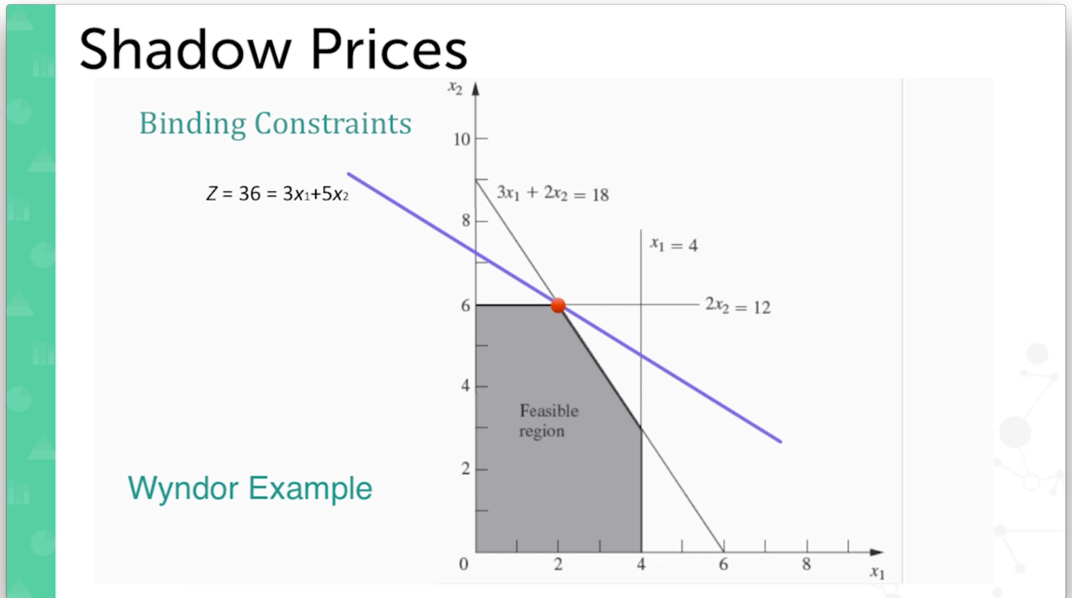
- **Shadow Prices:** The rate at which $Z$ is increased for each unit increase in the amount of resource $b_i$ (when $b_i$ remains in its *allowable range*)
- **The *marginal value* of a resource:** How much you should be willing to pay for additional units of a limited resource.
- **Binding Constraints:** The constraint is "=" at the optimal solution and the shadow price is greater than 0.

Shadow Price > 0 → Sensitive Parameter

A sensitivity report will show shadow prices associated with each constraint. The shadow price is the change in the optimal value of z for each unit increase in the constraint boundary b sub i, as long as the b sub i remains in its allowable range. If the b sub i has moved beyond its allowable range, then the current optimal solution will no longer be feasible.

Another way of describing this is to say that it's the marginal value of the resource associated with that constraint. The shadow prices guide managerial decisions about final resource allocation. Mathematically, the shadow prices are the optimal solution to dual problem. For this course, we'll concentrate on the interpretation of the shadow prices in the context of the sensitivity analysis.

When analyzing the right-hand sides of the constraints, the b sub i's that represent the amount of resource available, a shadow price that is not equal to 0 indicates a sensitive parameter. Its value cannot be changed without changing the optimal solution. This also means that the constraint for which that b sub i is the right-hand side of is a binding constraint. Meaning that it holds with equality at the optimal solution. The right-hand side is not sensitive if it has a shadow price equal to 0.

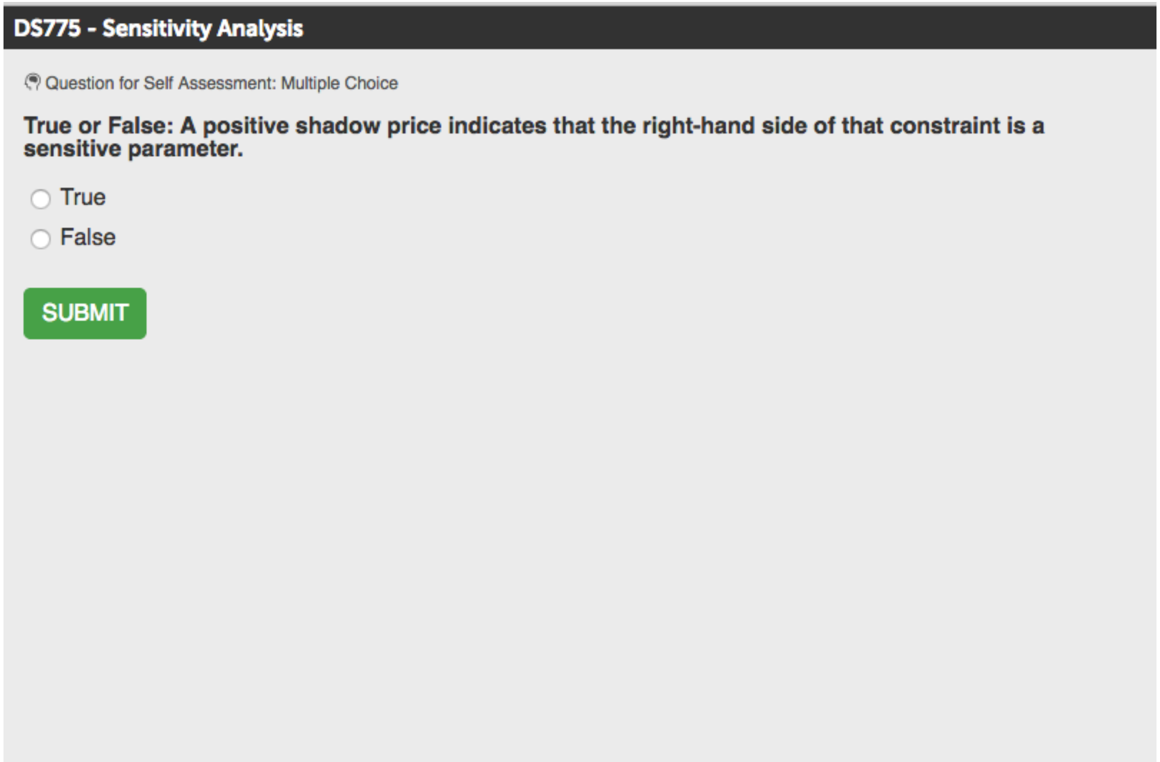



Here, we see the graphical solution to the Wyndor Glass example that's so often used in this book. Now, as the objective function comes through and finds its optimal point here, we see that the optimal solution is that z is 36. And that occurs where x1 is 2, and x2 is 6. Where this red dot occurs.

So you can see, this is at the intersection of constraints 2 and 3 in this problem. And so if either one of those changed-- if the right-hand side of either of those constraints changed, it would move that, and that would definitely have an effect on the optimal solution. And that effect is reflected in the shadow price for each.

Because that solution is found at the intersection of those two constraints, then these constraints are binding. Now, constraint number 1 over here is not part of the solution. And therefore, if it's moved, if its right-hand side changes, if it gets bigger or smaller, nothing happens to the optimal solution. And therefore, it has a shadow price of 0.

Question 1




Answer is at the end of this transcript

Question 2

**DS775 - Sensitivity Analysis**

Question for Self Assessment: Multiple Choice

**True or False: A positive shadow price indicates that the constraint is a binding constraint.**

- True
- False

SUBMIT

Answer is at the end of this transcript

## Sensitivity Report – Allowable Ranges

- For $c_j$ – the range of values for this coefficient over which the current optimal solution remains optimal, assuming no change in the other coefficients.
- For $b_i$ – The range of values for the right-hand side of constraint *i* over which the current optimal solution remains feasible and corresponding shadow prices are valid, assuming no change in the other right-hand sides.

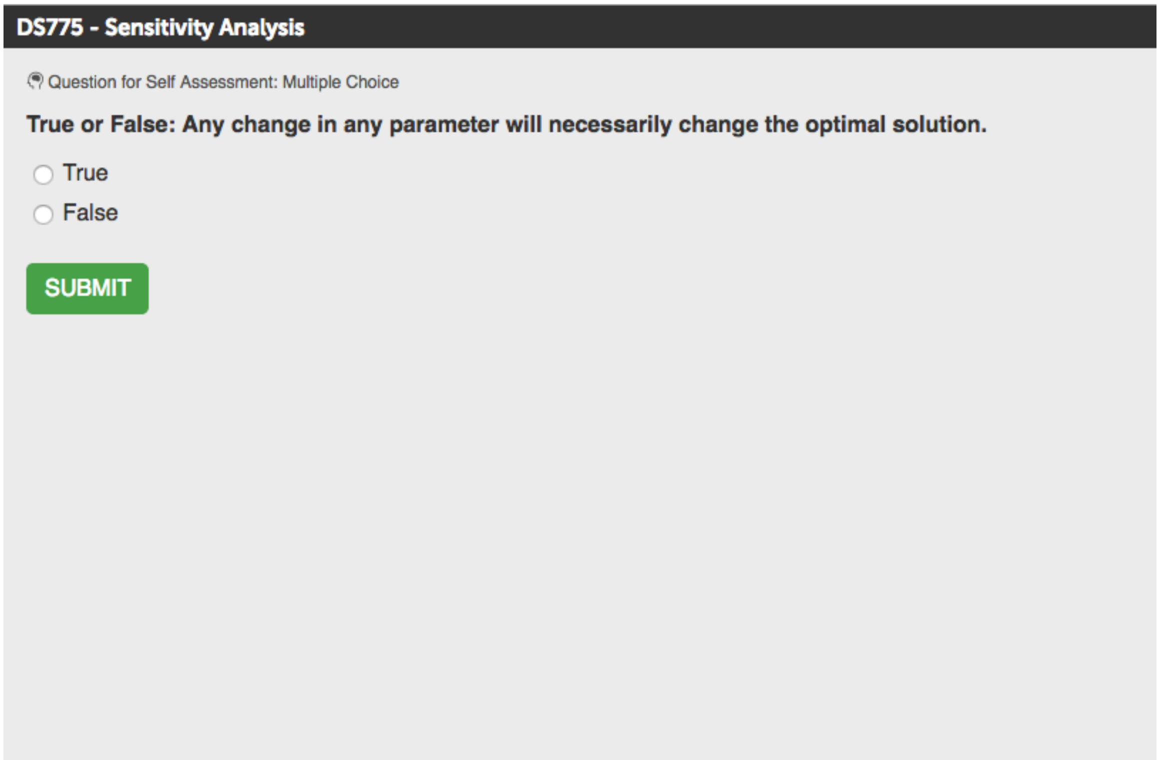
(Top table in Excel Sensitivity Report)

(Bottom table in Excel Sensitivity Report)

The allowable range for the coefficients in the objective function, the c sub j's, can be found from the top table of the Excel sensitivity report by looking at the allowable increase and allowable decrease for each one. The allowable range for an objective function coefficient is obtained by subtracting the allowable decrease and adding the allowable increase to the initial value. This is the range of values of what the coefficient can be so that the current optimal solution remains optimal, assuming that no other coefficients are changed.

The allowable range for the constraint boundaries, the b sub i's, is found in the bottom table of the Excel sensitivity report by subtracting the allowable decrease and adding the allowable increase to the initial constraint boundary. Note that these constraint boundaries are labeled as the right hand side values in the Excel sensitivity report. The allowable range for the b sub i's are the values for which the current optimal solution remains feasible and corresponding shadow prices remain valid, assuming no change in the other right hand sides.

Question 3




Answer is at the end of this transcript

# Kibbutzim Example 1 – Sensitivity Report

VIDEO

# Kibbutzim Example 2 – Parameter Analysis

VIDEO

# Robust Optimization

- **Soft Constraints:** can be violated a bit without serious complications
- **Hard Constraints:** *must* be satisfied

- Finds a solution that is guaranteed to be feasible and nearly optimal
- Each parameter is assigned its worst case scenario value within the range of uncertainty
- Assume parameter values are not influenced by each other
- Functional constraints must be in the form of $\leq$ or $\geq$

When a constraint boundary is not known with certainty, a range of possible values may be known. When parameters defining the constraint boundaries are not known with certainty and fall within a range, it's important to consider whether the constraints are soft constraints or hard constraints. Soft constraints can be violated a little bit without serious complications.

In other words, as long as we're close to that value, everything should be all right. Our solutions should still be feasible. These things are possible-- no serious complications. Hard constraints, on the other hand, must be satisfied.

Robust optimization is specially designed for dealing with problems with hard constraints. By using the lower or upper value of the range of possibilities for the parameters, basically a worst case scenario approach is used to ensure that the solution obtained is feasible and nearly optimal. Since the uncertain parameter can take on a possible range of values, only constraints of the form less than or equal to or greater than or equal to are used.

## Chance Constraints

- Constraint boundaries are random variables
- Pick a value in the distribution of values the constraint can take on such that there is a high probability of the constraint being satisfied (i.e. soft constraints)
- Use when upper and lower bounds for parameters are uncertain (normal distributions cover all real numbers, for example)
- Assume the constraint boundaries are independent
- Chance constraint as a probability expression vs. deterministic equivalent

When we started talking about linear programming at the beginning of this course, the certainty assumption was there to simplify the situation and taken to be true. Reality, as we found out, is often such that the parameters in the model are not known constants. In robust estimation, we suppose that we know a range of possible values the parameters may take on and go from there. Even this is a somewhat simplistic approach. Some parameters in a linear program are likely to be random variables, and so we should treat them as such in our problem-solving approach.

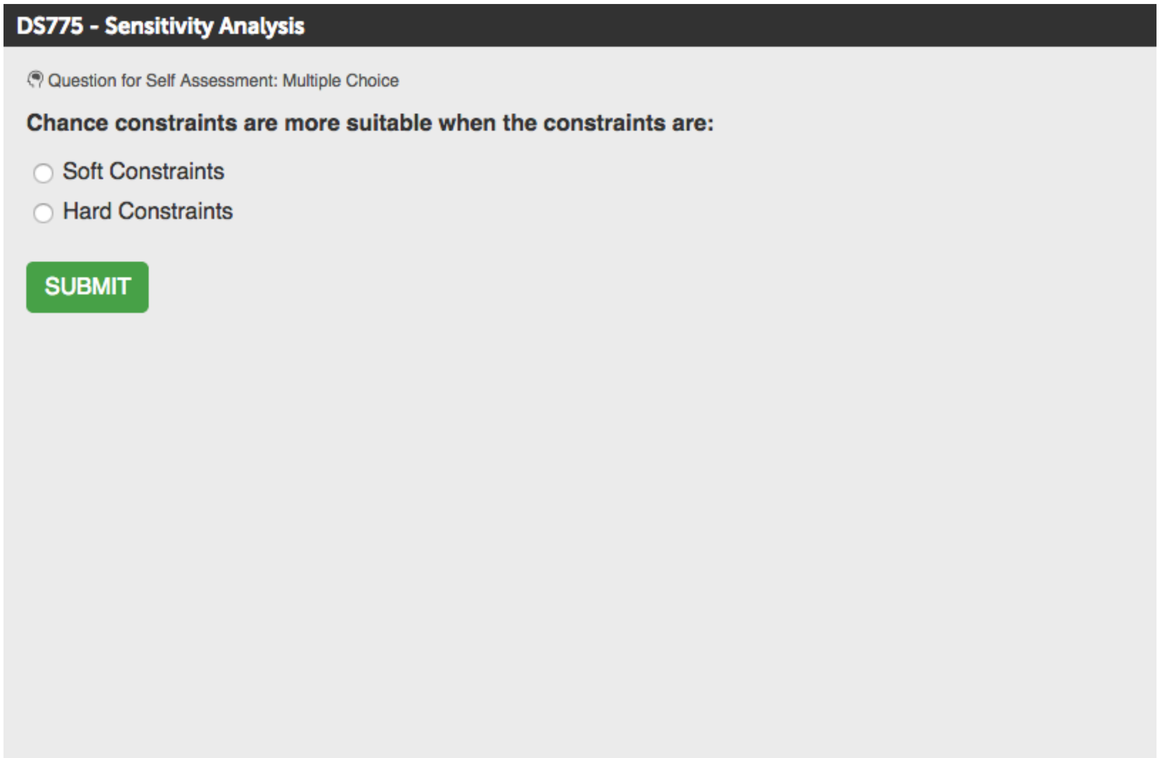
Suppose it is the constraint boundaries that are the random variables. Hopefully, these will be soft constraints rather than hard ones, because there will be a small probability of the constraint being violated when the outcome is implemented. The distributional family used to model the behavior of the chance constraint, along with its mean and standard deviation, are used to choose a value for the constraint such that there is a high probability of that constraint being satisfied.

Probabilities as highest point 95 and even higher are common choices. But this is up to the one conducting the analysis. When dealing with random variables, it is particularly helpful if we have a good idea of the probability distribution of the variables.

For better or for worse, the normal distribution is a popular choice among distributions used to model the behavior of parameters. It's also probably a familiar distribution and easy to work with. But other distributions can be used if they're more appropriate.

Basically, assuming independence of the random variables makes the math easier to compute joint probabilities of satisfying multiple constraints. A chance constraint can be expressed as either a probability statement or as a deterministic statement that is equivalent by selecting the appropriate percentile from the distribution of the constraint. This will be demonstrated in the coming example.

Question 4




Answer is at the end of this transcript

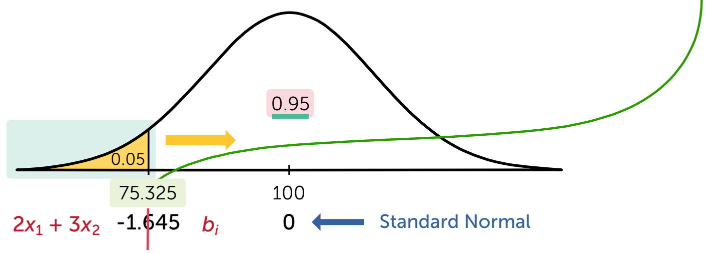
## Working with "Less-than-or-equal-to" Chance Constraints

Suppose a RHS constraint boundary $b_i$ has distribution $N(\mu = 100, \sigma = 15)$

**Constraint:** $2x_1 + 3x_2 \leq b_i$

**Probability Expression:** $P(2x_1 + 3x_2 \leq b_i) \geq 0.95$

**Deterministic Expression:** $2x_1 + 3x_2 \leq 100 - 1.645 \cdot 15 = 75.325$




Use `qnorm(p,0,1)` in R to look up values for $K_\alpha$ (e.g. $K_{0.95}$ = qnorm(.05,0,1))

When faced with a right hand side constraint boundary that's a random variable, select a value from the distribution of the variables that will result in a high probability of that constraint being satisfied. In this example, let's say that the parameter b sub i, a constraint boundary, is a random variable characterized by a normal probability distribution with a mean of 100 and a standard deviation of 15. Suppose the constraint is as shown here. 2 x1 plus 3 x2 must be less than or equal to the variable b sub i. It's important to remember here that the b sub i is a random variable.

The probability expression for this constraint looks like this. The probability that the constraint is satisfied is at least 0.95. The expression in here is an event with 2 x1 plus 3 x2 being taken as a constant and b sub i is the random variable. Using the appropriate critical value from the normal distribution that corresponds to the upper 95% of the distribution, the value from the distribution of b sub i can be computed and used to set up a deterministic expression that can be used in the linear programming problem.

In a standard normal distribution, the mean is 0 and the standard deviation is 1. The value that corresponds to the fifth percentile, which has an area of 0.95 above it, is negative 1.645. This is annotated as K sub alpha in the textbook. This is converted to a value of 75.325 in the normal distribution with a mean of 100 and standard deviation 15.

So if the expression 2 x1 plus 3 x2 is constrained to be less than or equal to 75.325, then it must fall in this region on the distribution. And so there's a 95% chance of the random variable b sub i being greater than that. Appendix 5 in your textbook has a table for you to look up the values of K sub alpha. But they can easily be found using R with the qnorm function as shown here.

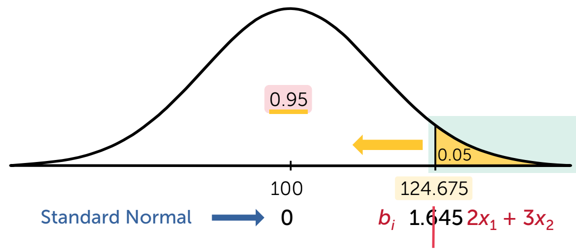
## Working with "Greater-than-or-equal-to" Chance Constraints

Suppose a RHS constraint boundary $b_i$ has distribution $N(\mu = 100, \sigma = 15)$

**Constraint:** $2x_1 + 3x_2 \geq b_i$

**Probability Expression:** $P(2x_1 + 3x_2 \geq b_i) \geq 0.95$

**Deterministic Expression:** $2x_1 + 3x_2 \geq 100 + 1.645 \cdot 15 = \underline{124.675}$




This time, let's suppose the constraint is a greater than or equal to constraint. That being said, the probability expression is that there is at least a point 95% probability of 2 x1 plus 3 x2 being greater than or equal to b sub i. Similar to the previous example, but on the other end of the distribution, we want to specify a value in the distribution so that there's a 95% chance of a random value of b sub i being less than that. That's another way of saying that the expression 2 x1 plus 3 x2 is greater than or equal to b sub i.

The cutoff value here is found this time by adding the critical value 1.645, which corresponds to the 95th percentile in the standard normal distribution, times the standard deviation 15 to the mean of b sub i, which is 100, to get 124.675. So then the deterministic expression can be written with 124.675 as the right hand side of the constraint. This way, when 2 x1 plus 3 x2 is constrained to be up here above 124.675, then there is at least a 95% chance that the random value of b sub i, when it is realized, will be below it.

# Stochastic Programming with Recourse

VIDEO

# IOR Tutorial – Graphical Solution and Allowable Ranges

VIDEO

# Sensitivity in OPL

VIDEO

# OPL Interactive Optimizer

VIDEO

## Question 1 Answer

Feedback for Self Assessment

✓ Correct!

**True or False: A positive shadow price indicates that the right-hand side of that constraint is a sensitive parameter.**

**Your answer:**

True

**Correct answer:**

True

## Question 2 Answer

**DS775 - Sensitivity Analysis**

Feedback for Self Assessment

✓ Correct!

**True or False: A positive shadow price indicates that the constraint is a binding constraint.**

**Your answer:**

True

**Correct answer:**

True

## Question 3 Answer

**DS775 - Sensitivity Analysis**

Feedback for Self Assessment

✓ Correct!

**True or False: Any change in any parameter will necessarily change the optimal solution.**

**Your answer:**

False

**Correct answer:**

False

## Question 4 Answer

**DS775 - Sensitivity Analysis**

Feedback for Self Assessment

✓ Correct!

**Chance constraints are more suitable when the constraints are:**

**Your answer:**

Soft Constraints

**Correct answer:**

Soft Constraints

**Feedback:**

Yes, since there will be a small probability of them being violated.