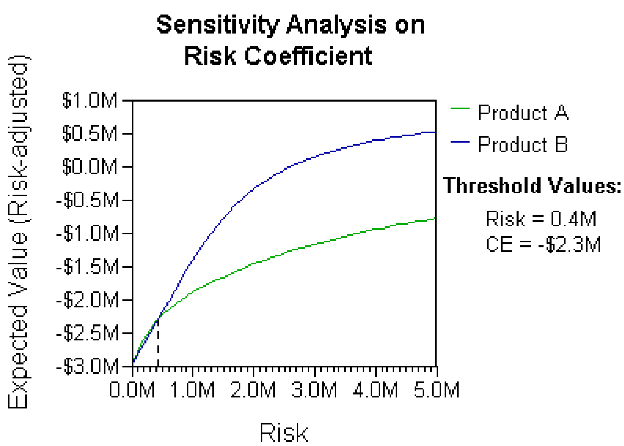
Southern New Hampshire University

DAT 520 Module Seven Overview

Sensitivity analysis is akin to a leave-one-out style of testing a model's robustness. Imagine that you have a bunch of variables and you leave one out to see if your results stay the same. Or imagine that you do not have very many variables, but you vary the values for one of them through a range, seeing how much the models results change. Or imagine sweeping two of those variables through a range simultaneously and looking for the threshold values where they cross. All of these examples are varieties of sensitivity analysis, which help you search for thresholds and inflection points in your model. You can also use them to detect errors and find ways to simplify your model.

The image below shows an example of a two-way sensitivity analysis. The dotted line indicates the threshold where the lines cross. At that threshold, the value of the blue product overtakes the value of the green one:



The other good thing about sensitivity analysis is that it lets you test your model for various inputs if there is uncertainty about what those values actually are. This can help you when you are trying to convey the meaning of your model to decision makers. If you can put your utilities through a range of plausible values, then you can express a rational range of the final values that come out of the model. Sensitivity analysis allows you to have a fuller conversation about what is in the model.

In this module, you are going to do more with Rattle. In Rattle, **minsplit**, **minbucket**, and **cp** are your main handles on how the model is constructed. You can track the improvement of your model by examining changes in the root node error, and how the xerror changes every time you adjust the model. Since you read Chapter 8 of the Larose text in Module Four, you understand how rpart works, and now you can implement and tune it more easily with Rattle.

The main driving question about sensitivity analysis and model diagnostics is: "What do you do with the information once you have it?" Typically you can report the ranges that you put the specific variables through, as well as the ranges of your resulting values. Or in the case of bottom-up modeling, you could talk about how you tuned the model to achieve the lowest error rate and best groupings of the variables. You may state why you tested and adjusted certain variables and not others, based on characteristics of those variables. You may want to talk about why you chose the ranges that you did for the testing. You may also talk about refinements to data collection in the future, based on the threshold values that you determined in your sensitivity analysis. There are many places for the information to land once you have put your model through its paces. The key thing is that you report what you did.

As far as bottom-up models, what you are looking to do is optimize them for the categories represented in the variables, as well as the variables in relation to one another. In later data mining classes, you will find that there are a lot more things that Rattle can do; for example, it can create random forests that are similar to sensitivity analyses because they vary the components of the model and give you a large set of possible models to choose from as the result, organized by those with the least error. We are not going to be covering random forests in this class; instead what we will do is concentrate on generating and tuning simple bottom-up decision trees based on discrete data sets. A lot can be learned from this activity that you can apply in future courses, such as predictive analytics.