# MINIMOD Robustness Tests for the Lives Saved Data

This document outlines the plans for how we will use the MINIMOD optimization tool to run robustness checks on the “Lives Saved” benefits data compiled by LiST and associated cost data.

## Brief Refresher

The MINIMOD Optimization tool solves the optimization problem that asks: “What are the optimal set of nutritional interventions across space and time, given their benefits and costs, and other policy relevant constraints?” By “policy relevant constraints” we mean constraints about the nature of the intervention (such as needing to be implemented nationally or having upfront costs before reaping benefits) and other constraints that might be important for stakeholders.

The solver was first written in GAMS and is now being finalized in Python. Both use mixed integer programming to find solutions to the optimization problem pose above. Given the nature of mixed integer programming as well as our responsibility to make robust and confident policy recommendations to stakeholders, it is important to test the sensitivity of our results.

## Sensitivity Analysis of Lives Saved Data

The factors used in the sensitivity analysis can be broken down into two major categories, data related checks and solver related checks:

1. *Data Related Checks*

* We will assume that the high and low estimates for lives saved are bounds on a uniform distribution
* Constructing a distribution over costs (a uniform with bounds of 20% above and below costs)

We will plans include using these two distributions to construct draws of benefits and costs, and running the optimization over these draws (probably 1,000 times)

1. *Solver Related Checks*

Over the course of the last few months, we have been working on getting the Python MINIMOD tool ready to take over the work of the GAMS tool. One thing we have observed during this exercise is that optimal solutions can change depending on the version of GAMS, the particular solver used to find optimal solutions and the tolerances of each solver. Although the optimal nutritional interventions do not change because of these differences, there are small discrepancies on *when* and *where* to implement them.

To make our results robust to these differences, we will also run sensitivity analyses on one key solver setting:

* Seed – The solver we use is probabilistic and uses a pseudo-random seed to make decisions. By changing this setting, we can test how global an optimal solution is and how robust it is to the algorithm used for optimization.

To test this, every draw from the date related checks will be run with 3-4 different seeds.

If you have any questions, please feel free to email me at [amichuda@ucdavis.edu](mailto:amichuda@ucdavis.edu)