

## 1 Introduction

Sensitivity analysis is an important part of assessing model behaviour and reliability. Sensitivity analysis looks at how the output of a model (i.e. simulation) varies, as the inputs are changed. The term *input* here refers to data that is fed into the simulation (primarily the `met` file), model parameters (e.g. saturated water content), and initial constants (e.g. the mass of initial surface residue). Sensitivity analysis can be used to rank inputs according to their influence on output variability, or screen out inputs which have little or no influence on the output. The results of a sensitivity analysis can also be used to justify the choice of input values, or to calibrate the model. The importance of a robust sensitivity analysis is further discussed in [Saltelli et al., 2010](#).

Many different sensitivity analysis methods exist. [Pianosi et al., 2016](#) provide a good high-level overview of various method types. Most methods follow the same basic algorithm:

1. Calculate the output for a range of input values.
2. Compare output variation relative to input variation.

The Morris method ([Morris, 1991](#)) allows the user to identify influential and non-influential numeric inputs. It is useful for screening out inputs that have a negligible contribution to output variability, regardless of the input value. Such inputs can be set to a constant value and excluded from more computationally intensive sensitivity analyses. It is also useful for identifying inputs that have a large contribution to output variability. Finally, the Morris method will reveal whether non-linear relationships or interaction effects are present (although it cannot be used to distinguish between the two).

The Morris method cannot be used to assign a proportion of output variability to each input. Variance-based methods are suitable for this (e.g. the Sobol method). The Morris method is also not suitable for use with categorical inputs or *composite* inputs. A composite input here refers to a collection of values that are typically not changed independently (e.g. a whole soil node, or `met` file). A factorial analysis of variance (ANOVA) allows quantitative analysis of categorical and composite inputs. It is however possible to run multiple Morris analyses (e.g. one for each soil type or `met` file), and qualitatively assess input effects (e.g. in different soils or climates).

This example demonstrates the use of the Morris method in the context of modelling nitrogen leaching from a cow urine patch. The simulation is run for a winter-deposited urine patch in Ruakura. Understanding the factors that affect nitrogen leaching may help reduce its effects, while maintaining milk production. First, the Morris method is explained in more detail, and some potential pitfalls are highlighted. A range of results is then presented, which show the type of insight that can be gained from a Morris analysis.

## 2 Morris Method Setup

### 3 Base Simulation

The Morris method is run from a `Morris` node. It can be found in `Models`, and should be added at the same level as the `DataStore`. The `Morris` node requires a base simulation. Expand the `Morris` node, `MorrisMethodExample`, to reveal the base simulation for this example (`UrinePatchSimulation`). The Morris method does not impose restrictions on the contents of the simulation.

The Morris method does require a specific reporting format however. Have a look at the `Report` node (expand the base simulation node and the `Field` node). Note that the report node must be named `Report`. The simulation report must also contain the year variable, `[Clock].Today.Year`, as it is expected by the `DataStore` functionality. The example report contains two other variables:

```
* [AddUrinePatches].Script.DepositionYear as DepositionYear
* [DoReportingSensit].Script.LeachN as LeachN
```

`LeachN` (nitrogen leached) is the reporting variable we are interested in analysing. In the context of sensitivity analysis, it is the *output* of the model. Additional outputs can be added, without increasing computational cost. Outputs must also be numeric.

If, like here, an output is reported several times, the Morris node will analyse each value separately. As you will see later, this allows you to get an idea of temporal variability in the sensitivity estimates.

If you select the `Clock` node, you will see the length of this simulation is 11 years (1972-1983). The reporting frequency for this example is set in a manager script (`DoReportingSensit`, found in the `Folder` node). `LeachN` is reported every two years, giving a total of five values for `LeachN`. These correspond to urine depositions in 1972, 1974, 1976, 1978, and 1980. You can view the urine deposition parameters in the manager script `AddUrinePatches`. Note that the reporting should not happen more frequently than annually (at present `[Clock].Today.Year` is used to group output into separate analyses). The reporting frequency should be set based on domain knowledge, and may correspond to a harvest year to investigate annual variability. `DepositionYear` is reported to allow us to plot the nitrogen leached for each deposition year, but is not needed for the Morris analysis.

The finished base simulation is copied into the Morris node. The original simulation may be deleted.

## 4 Morris Node

Now select the Morris node (`MorrisMethodExample`). Here you will find the various fields that need to be populated prior to a Morris analysis. Look at the bottom part of the user interface panel first. Here you add, as a separate row, each input which is to be analysed. In this example we are interested in the sensitivity of nitrogen leaching to soil organic carbon (`SOC`), nitrogen load within the urine patch (`Load`), and soil drainage (`DUL`). Each input requires the following:

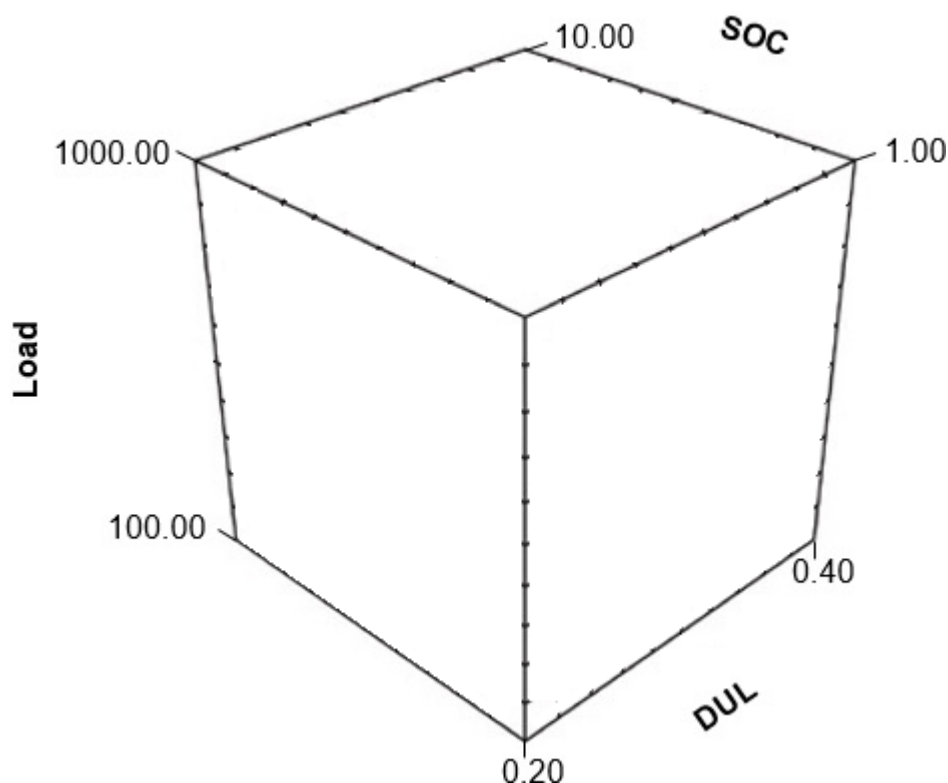
- \* `Name`: Used to identify the input in the analysis, but does not affect analysis results.
- \* `Path`: The location of the input within the simulation.
- \* `UpperBound`, `LowerBound`: The range over which the input will be sampled in the analysis.

The location of the input (`Path`) can be found with the help of `IntelliSense`. Try adding another copy of the input `SOC`. Note: all paths should start with the `Field` node. Entering the full stop character will bring up the `IntelliSense` dropdown box. The desired variable is part of an array (`OC`), so you need to finish the path with an index. Make sure to delete one of the `SOC` copies before running the analysis.

The fields `UpperBound` and `LowerBound` define the range over which the corresponding input will be sampled. This range is often considered to be the most important piece of information used to set up the analysis. In most cases, adjusting the input range will affect the results of the sensitivity analysis. Note this applies to not only the Morris method, but sensitivity analysis in general. Unfortunately there is no universal rule for selecting the range. The range should be selected based on domain knowledge, and should cover all values that you might reasonably expect to be model inputs. See Section 4 of [Pianosi et al., 2016](#) for more information on defining the input range.

This may seem obvious, but the analysis will not identify important inputs if they are not explicitly included in the analysis!

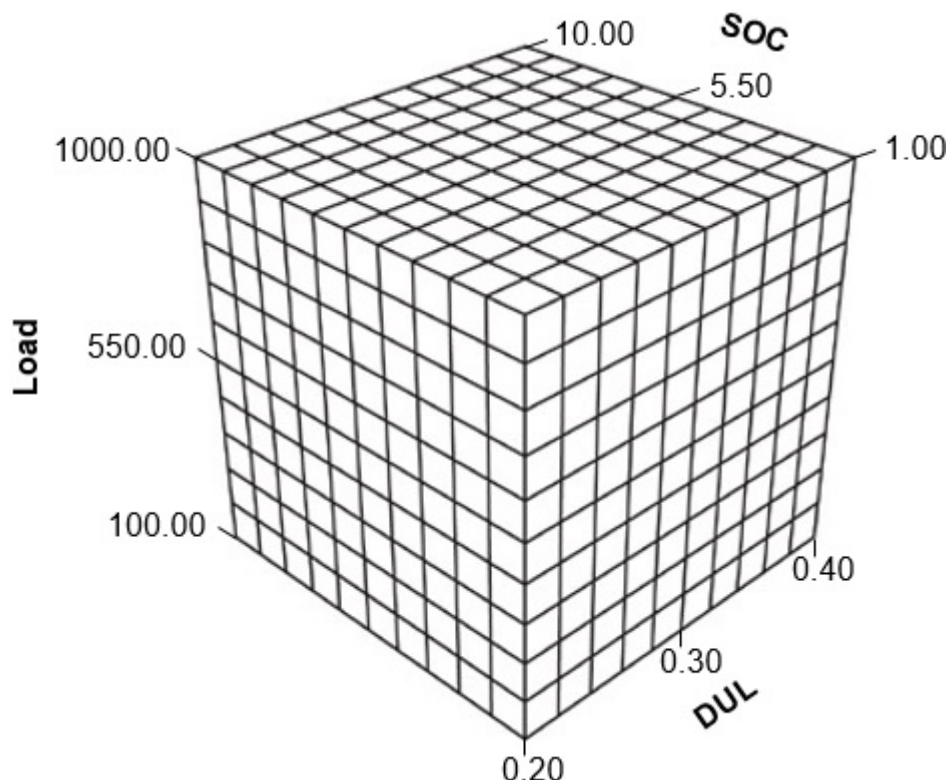
At this point we have defined the input space:



**Figure 1:** Sample Grid

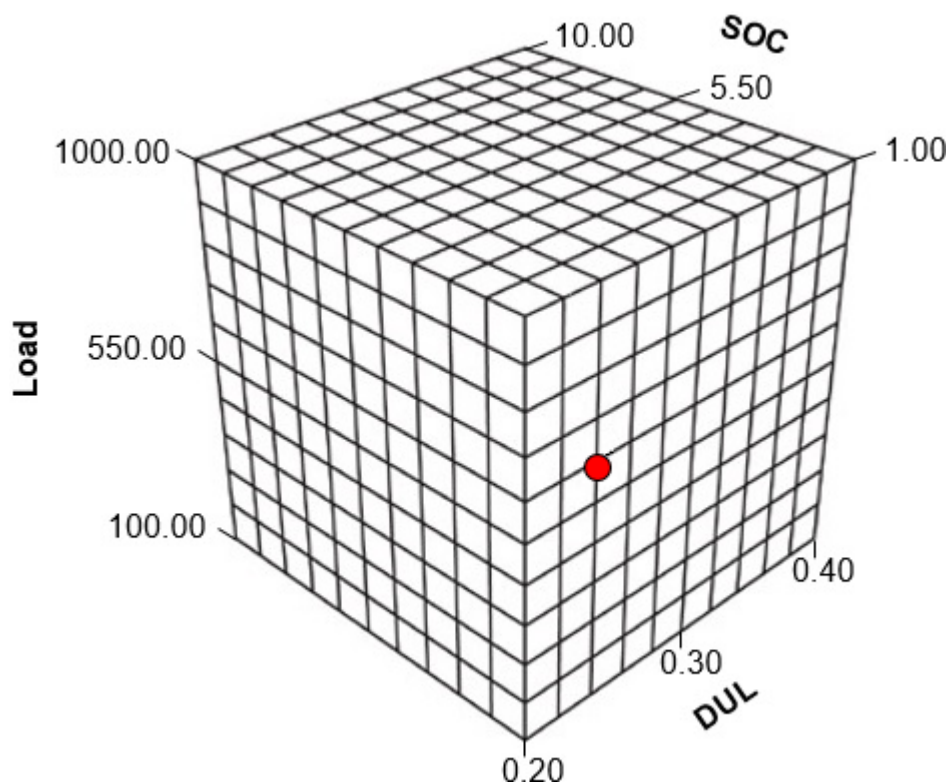
Now look at the top part of the user interface panel. Here are located a few more analysis settings:

\* **Number of intervals:** The number of intervals each input range is split into. In this implementation this is constrained to the same value for all inputs. The input sampling space is now a 3-dimensional grid, where each possible input value is called a *level*. The number of levels is the number of intervals + 1. The levels should adequately sample the input space - an input that is suspected to have a non-linear relationship with the output warrants a greater number of levels (and hence a greater number of intervals). The resulting grid:



**Figure 2:** Sample Grid

\* **Number of paths:** Note this does not refer to the location path of a variable, such as the one you entered above! It is the *number of samples* in the Morris analysis. From here on, we will use the term path exclusively in a sampling context. A greater number of paths will give more reliable results, but increase the analysis run time. Now imagine manually selecting a sample point:



**Figure 3:** Sample Grid

\* **Jump**: This is the magnitude of the change in each input, used to calculate sensitivity. It is an integer specifying the number of levels that each input will be changed by, from the base input value (red dot). The recommendation made by Morris is to set **Jump** to half the number of intervals. The jump is represented by an arrow in the grids below.

## 5 The Morris Method Algorithm

At this stage we have set up the analysis. Run APSIM from the **Simulations** node to start the analysis, if you have not done so already. It may take a couple of minutes to finish.

The Morris method will randomly select  $n$  points from the sampling grid. For each point (i.e. combination of input values):

1. The model output is evaluated.
2. For each of the  $m$  inputs:

\* The input value is increased or decreased by the number of levels specified in **Jump**. This selects a new point in the grid. Look at the grids below for an illustration, keeping in mind **Jump** is set to 5 for this example.

\* The output is re-evaluated for this new point.

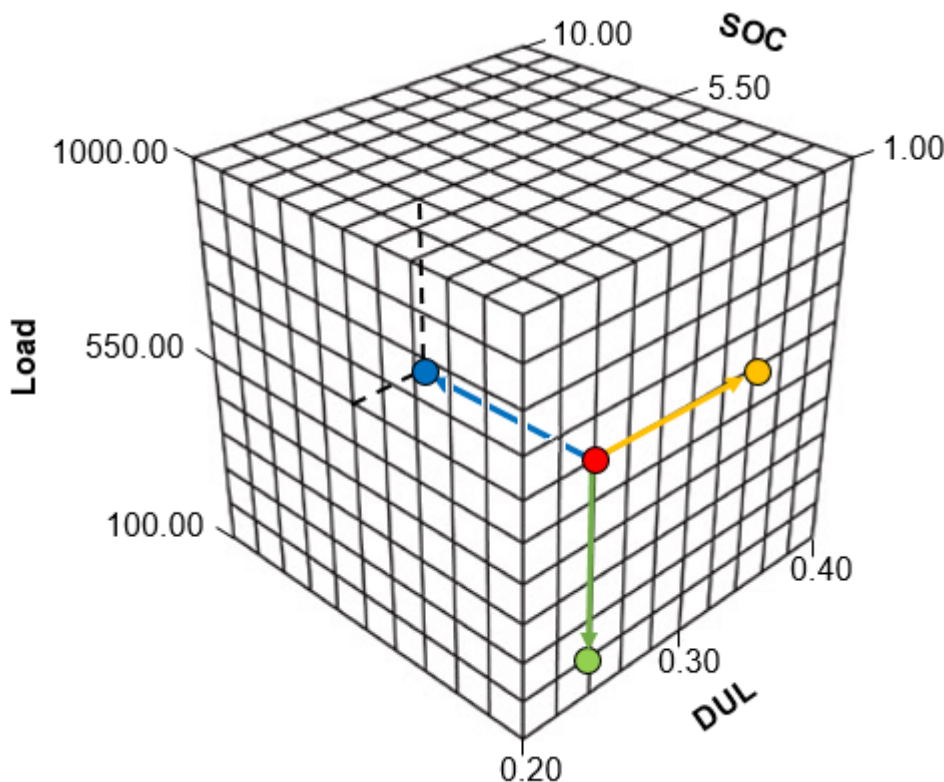
\* The input value is changed back to the initial value.

\* An *elementary effect*,  $EE$  (for the  $i$ th variable, at point  $x$  in the grid), is calculated by dividing the change in the output by the change in the input:

$$EE_i(\mathbf{x}) = \frac{y(x_1, \dots, x_i + \Delta, \dots, x_d) - y(\mathbf{x})}{\Delta}$$

**Figure 4:** Elementary Effects Equation

Jump illustrations:



**Figure 5:** Sample Grid

Note that the point selected after a jump is always inside the sampling grid. The requirement to stay within the sampling grid dictates whether the jump is an increase or a decrease. Now we are in a good position to understand what the term *path* really refers to. It is a sampling construct that encompasses:

- \* The sampled grid point (red dot).
- \* The corresponding set of  $m$  jumps (arrows).
- \* The set of  $m$  new points, derived using jumps (blue, yellow, and green dots).

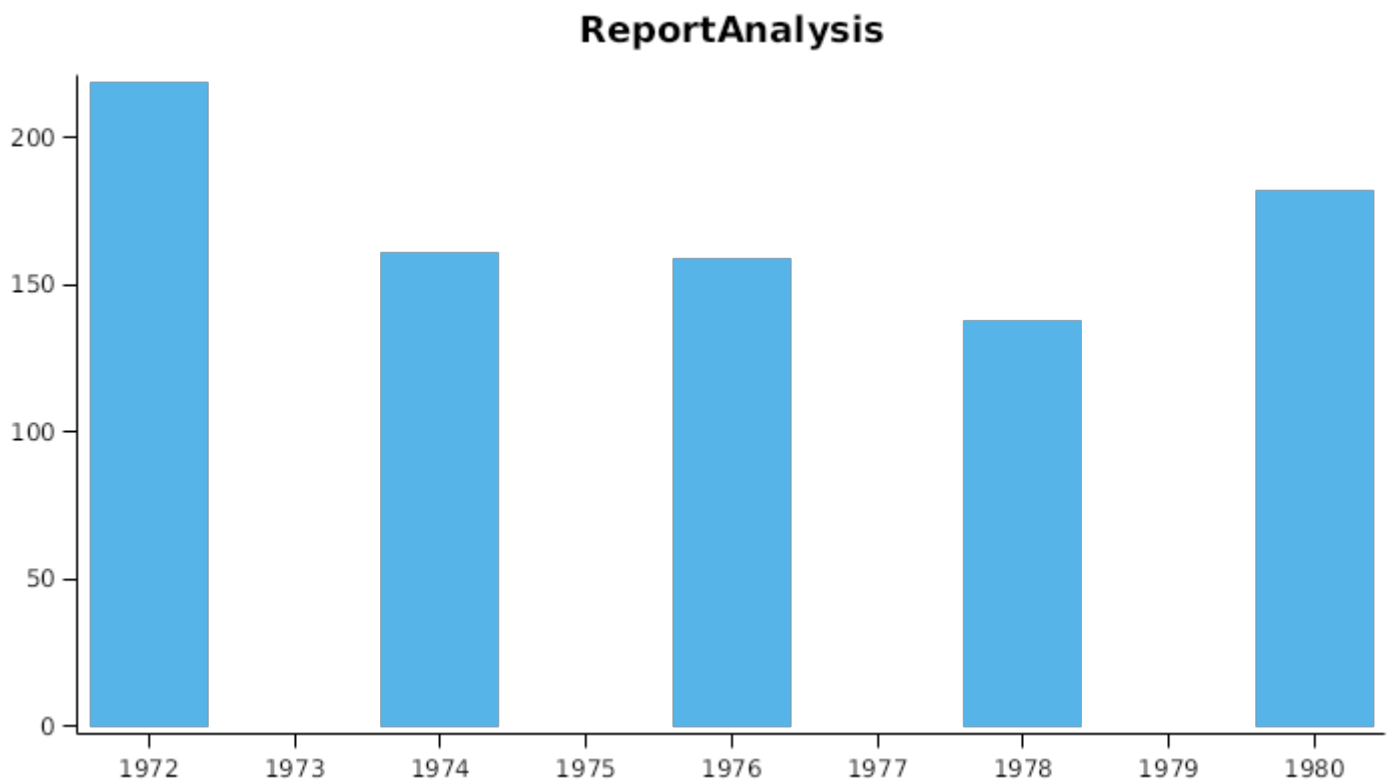
At this stage of the algorithm, all of the simulations have been run. For each of the  $m$  inputs we have  $n$  elementary effects. Each elementary effect is only a *local* measure of sensitivity however. This means it is a measure of sensitivity at one particular point in the input space. The sensitivity will usually change at different points in the input space however! This problem is solved by aggregating the  $n$  elementary effects for each input into an average (*mu star*) and a standard deviation (*sigma*). The average *mu star* is the mean of the *absolute values* of the elementary effects, as positive and negative effects may cancel out otherwise. It is a *global* measure of sensitivity, meaning it is derived from points that are spread all over the input space.

## 6 MorrisMethodExample

Encapsulates a Morris analysis.

## 7 Analysis Results

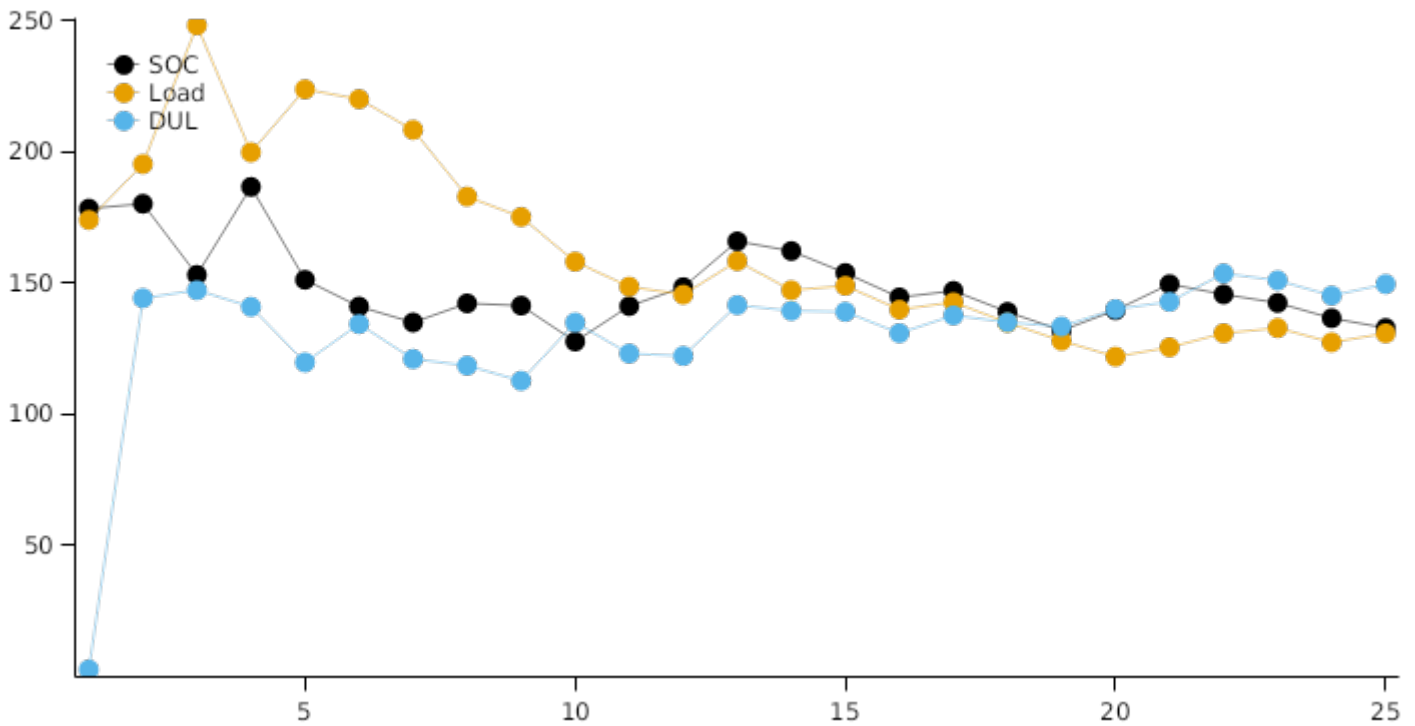
Once complete, the analysis produces three reports: `Report` (the base simulation report), a `PathAnalysis`, and `Statistics`. In this example the result are analysed using the graph node. Figure 1, in the `ReportAnalysis` node, shows an example of a graph produced using `Report` data.



The global sensitivity measure *mu star* is based on a limited number of samples (or paths), meaning it is an estimate. It is important to check the estimate has stabilised. Note: In this simulation we used a relatively small number of paths, to reduce analysis run time. This means the results may in fact not have stabilised and may differ between runs! You can check the estimates using the `PathAnalysis` data source. Select the `PathAnalysis` node. The provided graph (Figure 2) shows an estimate of *mu star* for each sample size, from 1 to  $n$  (where  $n$  is the set number of paths). Each data series should flatten out as the number of paths increases. If this is not the case, try increasing the number of paths in the Morris node.



## PathAnalysis



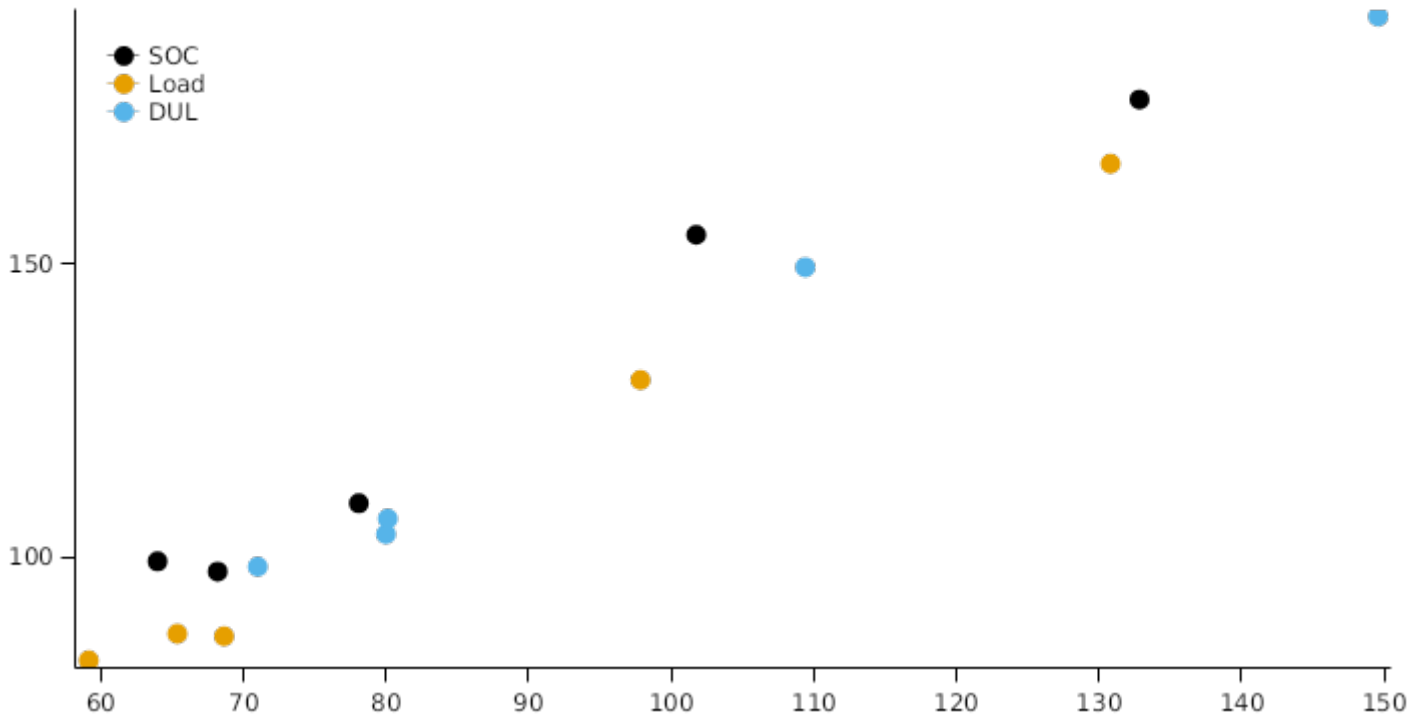
The `Statistics` data source holds *mu star* and *sigma* estimates for each input and output combination. A classic Morris method plot (with *mu star* on the *x*-axis and *sigma* on the *y*-axis) is shown in Figure 3. Select the `Statistics` node to view it. The plot contains five sensitivity estimates for each input (corresponding to five reporting years). The plot can be separated into four quadrants (note these are approximate and should only be used as a guideline):

- \* Bottom left: low *mu star* and low *sigma*. Inputs that have little influence on the output and may be fixed.
- \* Top left: low *mu star* and high *sigma*. Inputs that, on average, have little influence on the output, but have either a non-linear relationship with the output, or interactions with other inputs. Note that the Morris method cannot separate non-linearities from interaction effects.
- \* Bottom right: high *mu star* and low *sigma*. Inputs that are influential, have a linear relationship with the output, and have no interactions with other inputs.
- \* Top right: high *mu star* and high *sigma*. Inputs that are influential, but have either a non-linear relationship with the output, or interactions with other inputs.

The five different estimates for each input give an indication of temporal variability. The nitrogen load appears to be the most important input contributing to variability in nitrogen leaching. Soil organic carbon and soil drainage are also important in some years however. All significant inputs are subject to interaction effects or non-linearities. The large spread observed in each estimate indicates high temporal variability.

To get a better feel for the data produced by the analysis, try reconstructing the three graphs using a new graph node. You will need to select the correct data source and correct variables within that data source. Try using the `Filter` feature in the `Series` node to produce a less crowded plot.

## Statistics



## 8 Conclusion

The urine patch deposition simulation was used to demonstrate the process of performing sensitivity analysis in APSIM. The steps required to set up a Morris analysis were outlined, and results that can be obtained using current functionality were shown. The Morris analysis plot (Figure 3) is particularly insightful, as it highlights inputs that drive output variability. The plot in Figure 3 also highlights the large variability observed year to year. The Morris method is not computationally intensive, and hence often serves as a preliminary analysis. The Sobol method or factorial ANOVA can be used to obtain a more quantitative result, perhaps focussing on a smaller number of inputs.

## 9 References

- Morris, Max D, 1991. Factorial Sampling Plans for Preliminary Computational Experiments. *Technometrics* 33 (2), 161-174.
- Pianosi, Francesca, Beven, Keith, Freer, Jim, Hall, Jim W., Rougier, Jonathan, Stephenson, David B., Wagener, Thorsten, 2016. Sensitivity analysis of environmental models: A systematic review with practical workflow. *Environmental Modelling and Software* 79, 214-232.
- Saltelli, Andrea, Annoni, Paola, 2010. How to avoid a perfunctory sensitivity analysis. *Environmental Modelling and Software* 25 (12), 1508-1517.