# ACE – Application for Conditional probability Elicitation: Manual

## Folder Structure

Within the top level folder ACE, the R-shiny .ui and .server files are located along with two script files (createTable.R and plotTable.R) required for the app. Note, the name of this folder should not be edited due to following dependencies.

Within this top level folder the subfolder www contains the bootstrap css theme used to style the app.

An additional three subfolders store and maintain the elicited networks. The folder nodeTables contains the auxiliary files needed to store the structure of the networks to be elicited. Upon distribution, the folders initialCPTs and savedCPTs will be empty.

## Required R Packages

library(shiny)

library(shinyjs)

library(shinythemes)

library(rhandsontable)

library(plotrix)

library(grid)

library(igraph)

library(shape)

library(RColorBrewer)

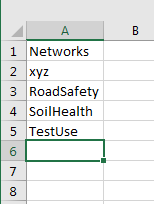
library(lattice)

library(latticeExtra)

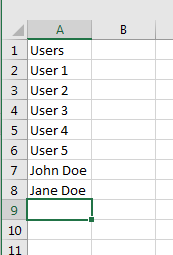
## Setup

To initialise the app with a new network, 4 auxiliary files in the subfolder nodeTables need editing.

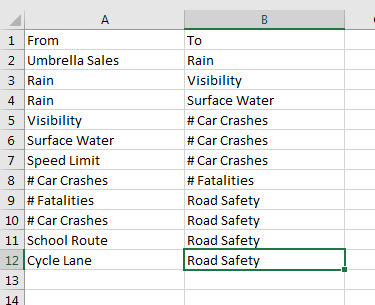
1. Add the name of the network into the file NetworkNames.csv



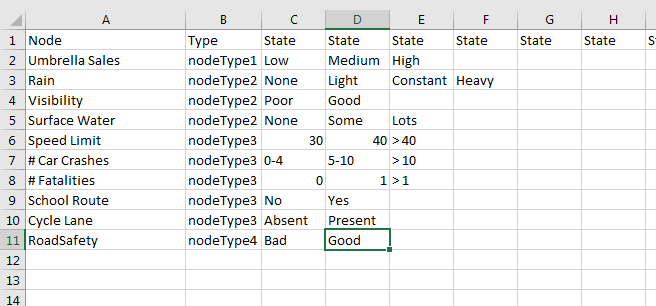
1. Add the list of users into the file Usernames.csv



1. For each new network, create a file containing the structure of the network. This should be stored in a csv file named Edge\_NetworkName.csv (e.g. Edge\_RoadSafety). This file encodes the structure of the network, where values in the first column (From) correspond to the parent node (or source) of the edge and the second column (To) the child node.



1. For each new network, create a file containing the definition of each node. This should be stored in a csv file named Node\_NetworkName.csv (e.g. Node\_RoadSafety.csv). This file captures the definition of the states of each node. Only categorical nodes can be defined, but there is no limit on the number of states allowed.



The first column contains the name of each node, the second column contains a categorisation of the type of each node. This is used in the visualisations but has no impact in the creation of the conditional probability tables. The subsequent columns are used to define the states of each node. There is no restriction on the number of states allowed, but it is recommended to make these states succinct but descriptive and listed in a logical order (most commonly a notion of “increasing” order).

1. Upon completion of the above 4 files, open R and run the file preliminaries.R (located in the top level folder).

Set the working directory to the app location, e.g.

setwd("D://")

source("ACE/preliminaries.R")

Upon running the preliminaries.R script, a number of additional files will be created in the subfolder /nodeTables/ including a .pdf of the network structure. The layout of this network is automatically generated from the force-directed layout algorithm by Fruchterman and Reingold (1991) using the layout\_with\_fr function of the igraph package.

In addition, the full set of initialised conditional probability tables will be stored in the initialCPTs subfolder. These are used to initialise and reset the main app and will not contain reusable data.

## Loading the app

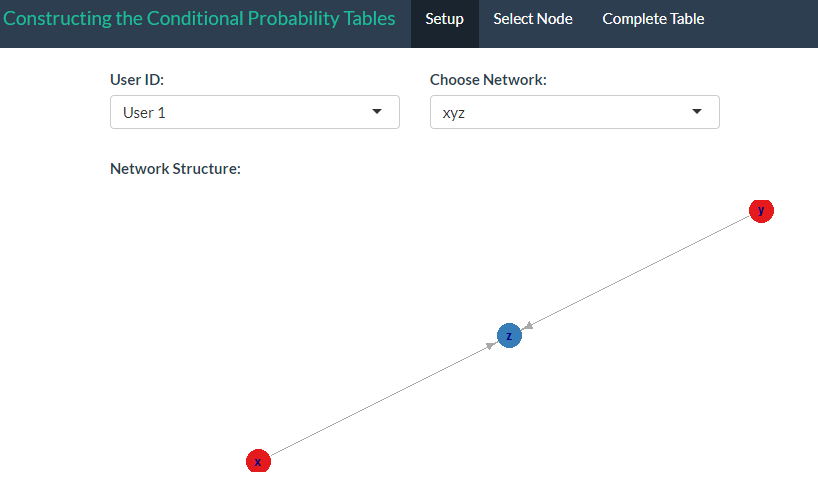
The app can run on a shiny-server or as a local shiny application. To launch the app locally,

runApp("ACE", launch.browser = TRUE)

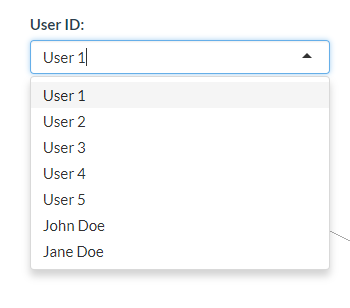
Note, we have found the app to be more stable when launched in an external browser compared to the R-shiny browser.

## Using the app

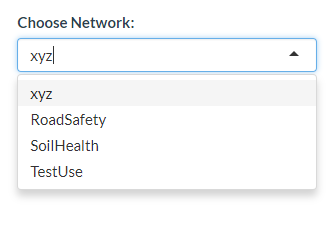
Upon loading the app, you should see a preview of the default Network



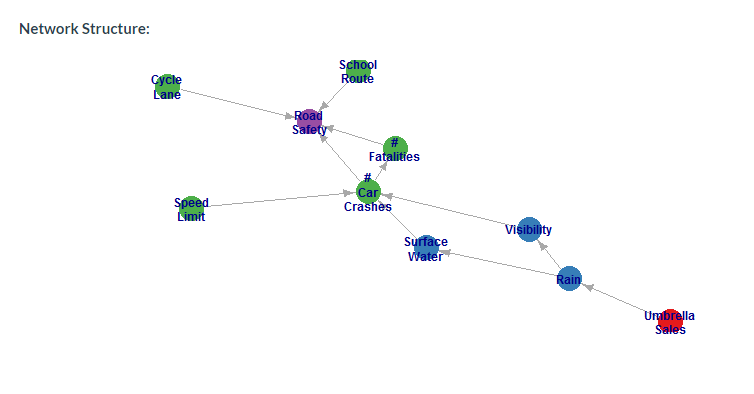
The User can select an alternative Username from the dropdown list (corresponding to the list of names in Usernames.csv)



The Network of interest can be selected from the right hand drop down list (corresponding to the list of networks in NetworkNames.csv)



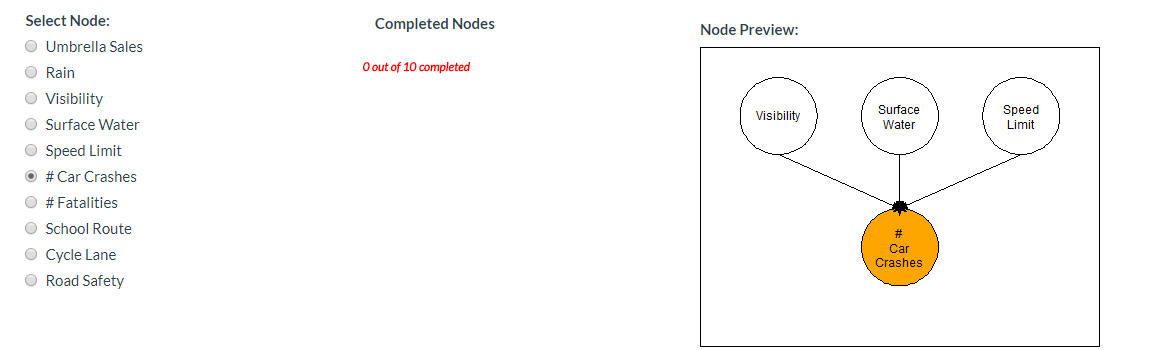
Upon selecting the network of choice, e.g. RoadSafety, a full preview of that network diagram should be visible.



Once the appropriate network and username has been selected, the user can move to the next page, through the, “Select Node” option of the main menu.

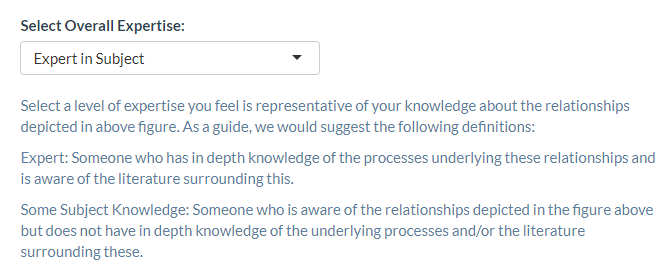


From this page, each individual node can be selected, from which a preview of the conditional dependencies are shown.



A list of the completed conditional probability tables (CPTs) will be shown in middle column (note, this only gets updated when a user selects a new node).

From this page, the user can select a level of their expertise about the selected node.



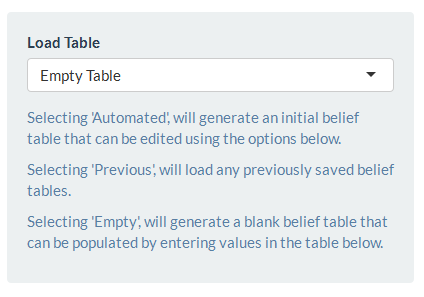
This information is not directly used in the formulation of the conditional probability tables, but rather can provide auxiliary information when the resulting networks are combined from multiple experts or as part of a sensitivity analysis.

Once a node has been selected, the user can move to the Complete Table menu

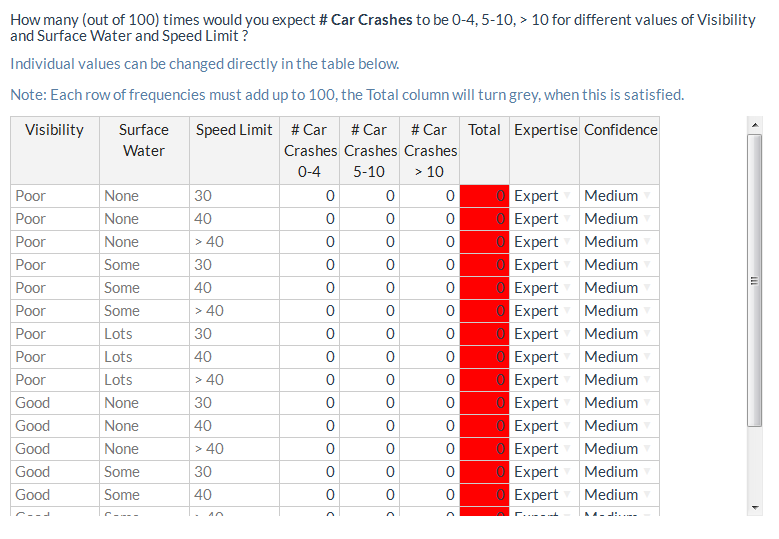


It is through this page that users quantify their belief about the distribution of each node through a conditional probability table.

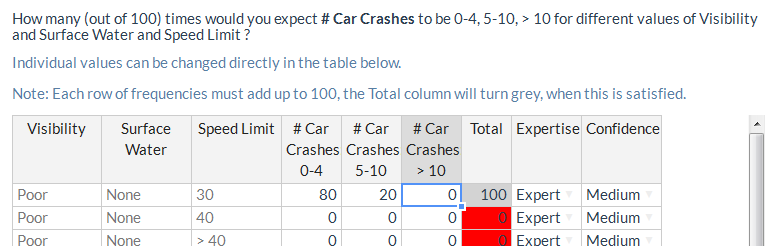
In its basic form, the app provides a simple interface to filling out a CPT using a frequency format, where users are asked how many times out of 100 they would expect the child node to be in each of its possible states for every combination of parent node states. To do so, a user must select the “Empty Table” option from the Load Table menu.



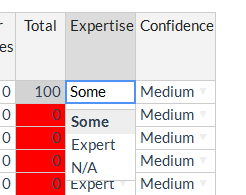
This will generate an empty conditional probability table as shown below.



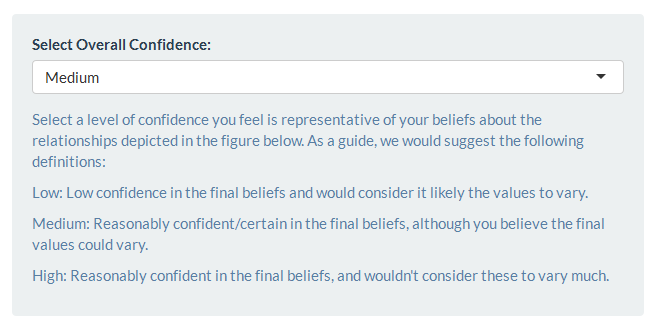
This table is fully editable, where each row corresponds to a single probability distribution of the child node. For example, the first row looks to define the distribution of # Car Crashes when Visibility is Poor, there is No Surface Water and the Speed Limit is 30. The distribution is defined in the three columns labelled “# Car Crashes” and once these values sum to 100, the “Total” column will turn grey.



The “Expertise” column contains the level of expertise the user had selected on the previous page. These values can be edited for individual entries from the associated dropdown box in each cell.



The “Confidence” column records the level of confidence a user has about their elicited distribution. This can be set for the entire table globally through the dropdown box.



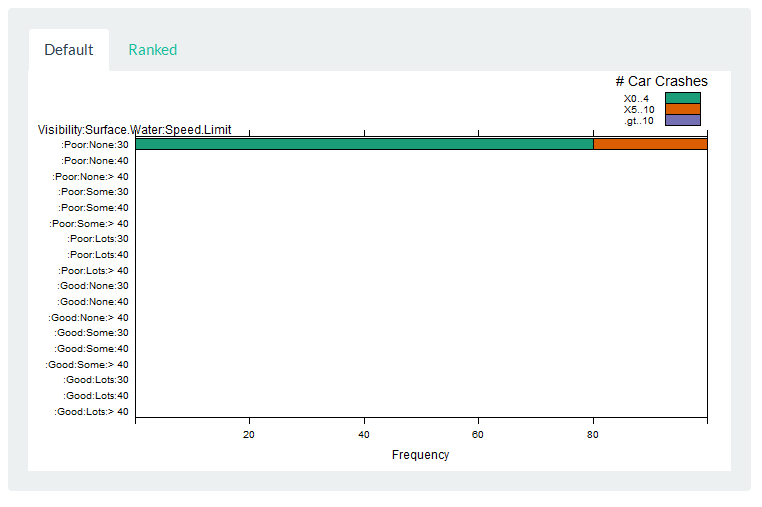
And can be edited for individual table entries from the associated dropdown box in each cell.



This information is not directly used in the formulation of the conditional probability tables, but rather can provide auxiliary information when the resulting networks are combined from multiple experts or as part of a sensitivity analysis.

To aid the numerical elicitation, a graphical representation of the CPT is provided by clicking the “Update Visualisations” button.





Note, this visualisation does not live update and can only be refreshed by clicking the “Update Visualisation” button.

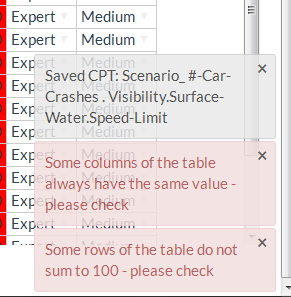
Once a user is happy with the conditional probability table, clicking “Save CPT” will write the conditional probability table into a csv file, saved in the savedCPTs subfolder.



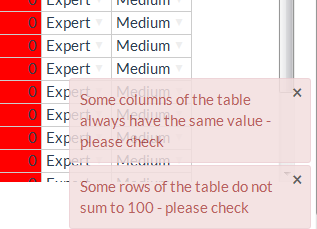
The resulting files having the following naming structure:

NetworkName\_ChildNode\_Parent1.Parent2.Parent3\_Username.csv

Upon successfully writing the CPT table, a grey pop-up message will appear briefly on the screen confirming the file has been saved.

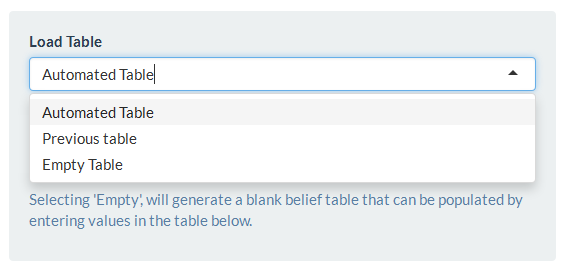


Various checks are performed on the CPT when it is saved, with suspicious entries highlighted in different pop-up boxes, e.g.

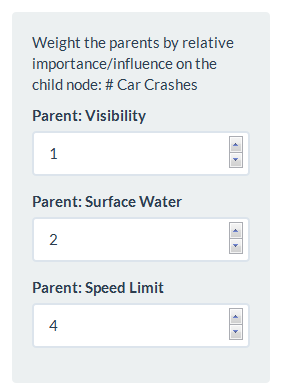


These messages do not prevent a successful save, but are instead aimed to inform the user they may need to check their entries and to edit if necessary.

Furthermore, this elicitation app enables an automatic approach to prepopulate the conditional probability tables. To enable this feature, select the “Automated Table” option from the “Load Table” menu.

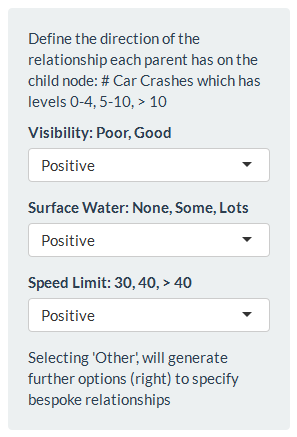


Upon selecting this option, two additional panels should become visible. The first panel allows the user to assign a relative weight to each parent node. Parents with a larger weight are assigned a greater level of influence in determining the conditional probability table such that changes in the states of the parent with the largest weight will result in the biggest differences in the distribution of the child node. In the example below, a change in the state of Surface Water will have an effect twice as large as a change in the state of Visibility.



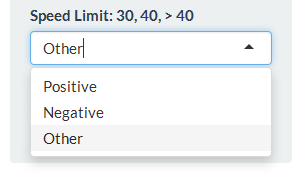
The weights must take integer values, but there is no upper limit. If all parents are deemed equally important/influential, a user should select the same weight for each parent.

The second panel enables the user to define the direction of the relationship between each parent and child.

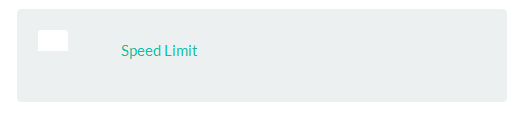


Each parent can have either a Positive, Negative or Other relationship with the child node. A relationship is considered positive if, as the states of the parent increases\* [according to the order they have been defined], the probability the child node is in its higher states also increases. Conversely, a negative relationship is appropriate if as the states of the parent increases\* [according to the order they have been defined], the probability the child node is in its higher states decreases.

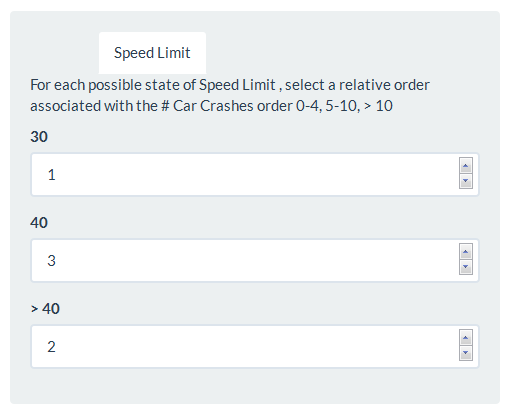
Not every parent-child relationship can be categorised as having either a positive or negative relationship. It is also impractical to incorporate a full set of relationships. Instead, incorporated into the app is the option to define an “Other” relationship.



Upon selecting “Other” from the dropdown box, a third panel becomes live. This enables users to define a “relative order” to the states of the parent node.

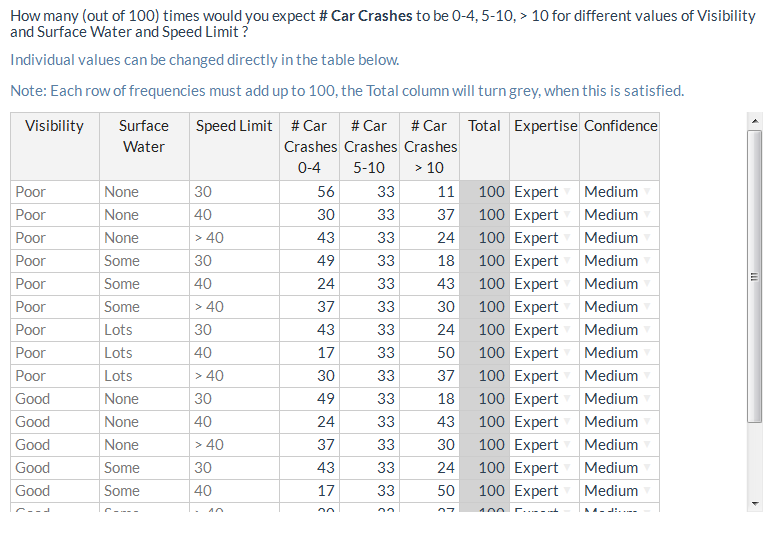


Note, not all options of this panel may be visible at first. Selecting the name of the parent node (e.g. Speed Limit) will bring forward the options.

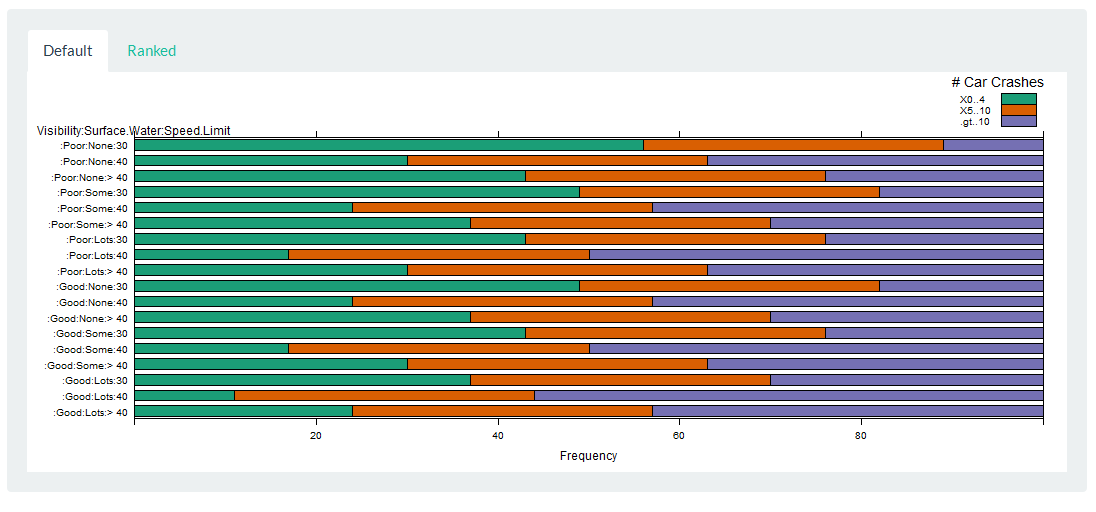


This relative order describes the order of the parent states that would result in an increasing probability that the child node is in its higher states. In the above example, a speed limit of 30 [order 1] is associated with the fewest car crashes, a speed limit greater than 40 [order 2] is associated with some car crashes and a speed limit of 40 [order 3] is associated with the most car crashes.

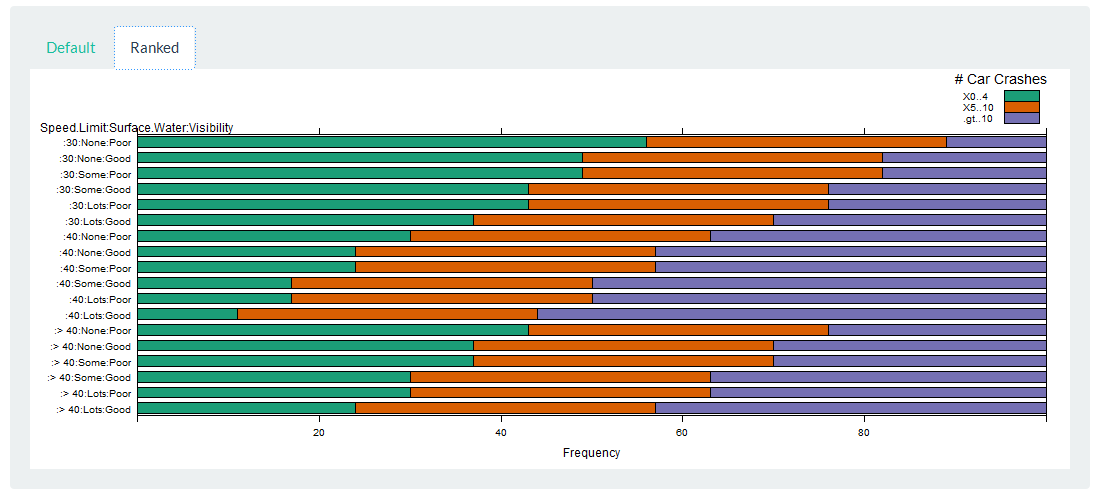
Mathematically, the relative weighting and order relationships define a score, detailed in Hassall et al. (in preparation) from which an initial draft CPT is created.



And can be visualised



By selecting the “Ranked” tab in the visualisation window, the graphical representation of the CPT is reordered according the relative weights defined for the parent set, thus providing a more intuitive display.

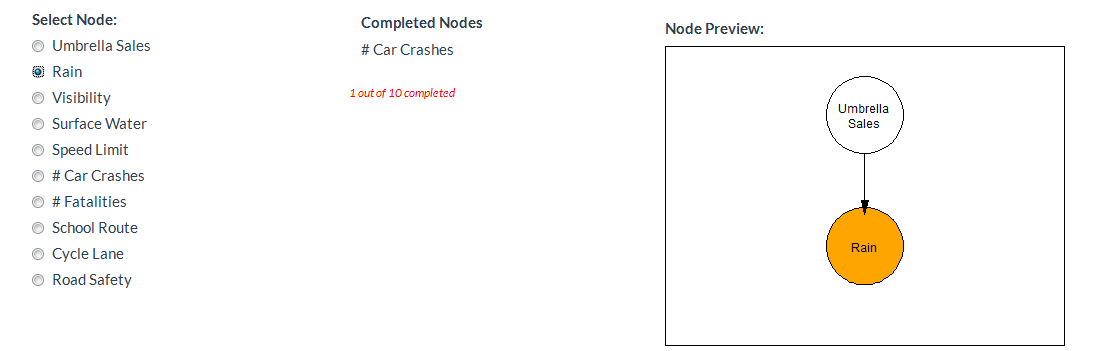


Note, this automated process is designed to prepopulate the CPT and a user is strongly encouraged to look at these values and continue editing individual scenarios.

Once a user is happy with their saved CPT, they should navigate back to the “Select Node” page

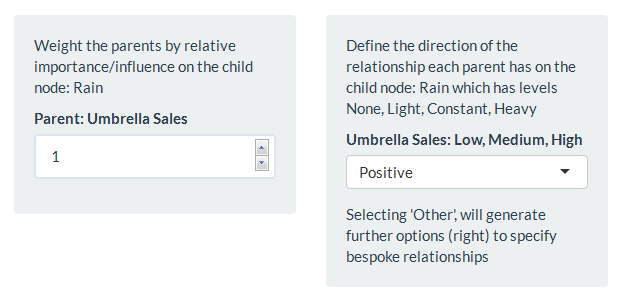


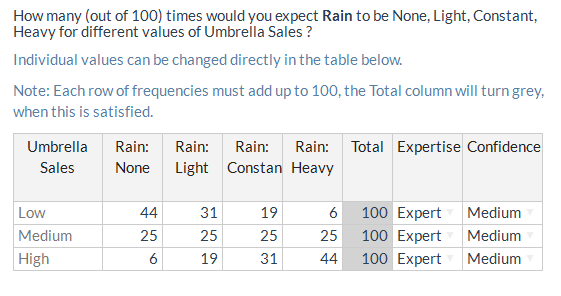
Upon selecting a new node, the user should see an updated list of the completed tables.

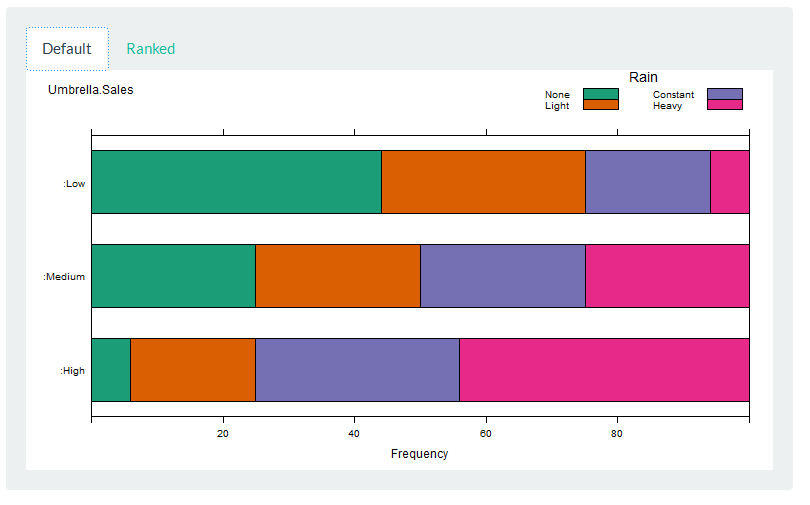


Navigating through to the “Complete Table” page, a new conditional probability table is ready to be completed.

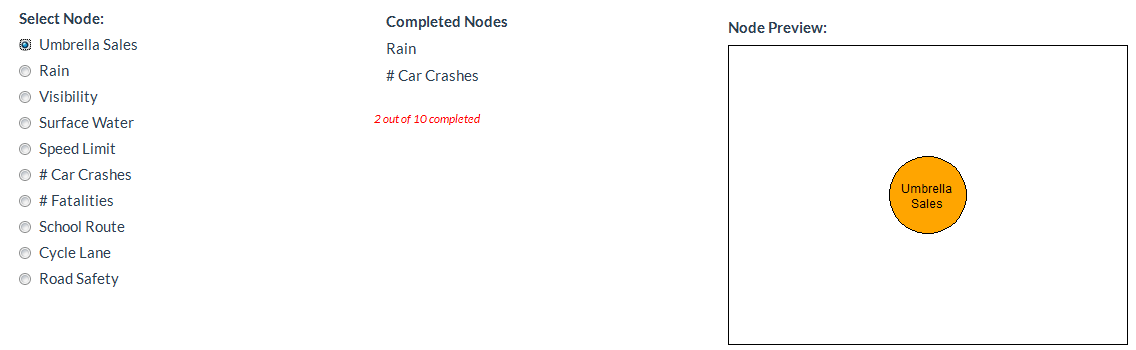
For a node with a single parent, the relative weighting will have no influence on the automatically generated table. However, a user can still define the direction of the relationship to prepopulate the table.



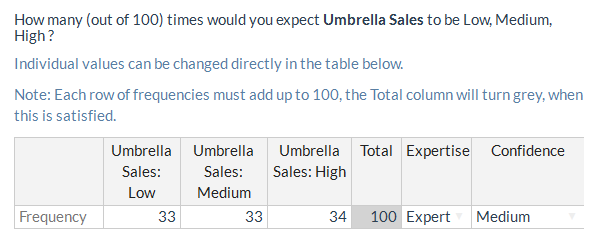




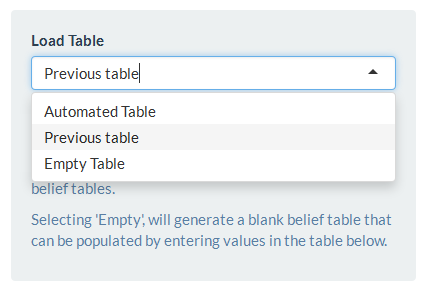
For nodes with no parents, e.g. Umbrella Sales, the conditional probability table corresponds to the overall distribution of that node. The app allows these distributions to be elicited from users, although in general the authors anticipate such nodes to be informed from data.



Selecting the “Automated Table” will generate a uniform distribution of the states of the node



A user can return to a previous node from the “Select Node” page. To load a previously saved table, select “Previous Table”.



This will load the previous CPT and allows a user to edit individual values in the table. However, the options used to generate this table are not recoverable.

When an existing table is edited and saved, this will overwrite (without warning) any previously saved tables.

## Common Questions

Q: Error when running preliminaries.R, (e.g. Error in as.igraph.vs(graph, v) : Invalid vertex names)

A: Check all node names are consistent across the Edge table (both To and From column) and the Node table.

Q: App crashes when saving CPTs

A: An external browser has been found to be more stable

Q: When saving the CPT, there is are spurious entries in the first column

A: These entries are used within the app to track changes to the CPT but are not relevant to the table entries.

Q: How large can I make the network?

A: The network can have as many nodes as needed. However, the app currently has a hardcoded limit that each child can have at most 6 parents. For R-shiny users, this can be manually edited in the .server and .ui files.

Q: How many states can a node have?

A: In most cases, there is no upper limit on the number of states for a node. However, if a user wants to make use of the “Other” option in the parent relationship definition, there is currently a hardcoded upper limit of 10 states. For R-shiny users, this can be manually edited in the .server and .ui files.

## References

Fruchterman, T.M.J. and Reingold, E.M. (1991). Graph Drawing by Force-directed Placement. Software - Practice and Experience, 21(11):1129-1164.

Hassall, K.L., Dailey, A.G., Zawadzka, J., Milne, A.E., Harris, J.A., Corstanje, R., Whitmore, A.P. (in preparation) Facilitating the elicitation of beliefs for use in Bayesian Belief modelling