SANDIA REPORT
SAND2007-1299
Unlimited Release
Printed March 2007

LinguisticBelief: A Java Application for Linguistic Evaluation Using Belief, Fuzzy Sets, and Approximate Reasoning

John L. Darby

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia is a multiprogram laboratory operated by Sandia Corporation,
a Lockheed Martin Company, for the United States Department of Energy's
National Nuclear Security Administration under Contract DE-AC04-94AL85000.

Sandia National Laboratories

SAND2007-1299
Unlimited Release
Printed March 2007

LinguisticBelief: A Java Application for Linguistic Evaluation Using Belief, Fuzzy Sets, and Approximate Reasoning

John L. Darby
Security Systems Analysis
Sandia National Laboratories
P.O. Box 5800
Albuquerque, NM 87185-0757

Abstract

LinguisticBelief is a Java computer code that evaluates combinations of linguistic variables using an approximate reasoning rule base. Each variable is comprised of fuzzy sets, and a rule base describes the reasoning on combinations of variables' fuzzy sets. Uncertainty is considered and propagated through the rule base using the belief/plausibility measure. The mathematics of fuzzy sets, approximate reasoning, and belief/ plausibility are complex. Without an automated tool, this complexity precludes their application to all but the simplest of problems. LinguisticBelief automates the use of these techniques, allowing complex problems to be evaluated easily. LinguisticBelief can be used free of charge on any Windows XP machine. This report documents the use and structure of the LinguisticBelief code, and the deployment package for installa-tion client machines.

Acknowledgements

John C. Cummings of the Advanced Concepts Group at Sandia National Laboratories was the sponsor of this effort. John provided much direction, significant advice, and proposed a technique useful for interpreting results.

The author appreciates the careful review of this report by Greg Wyss and Peter Merkle of Sandia National Laboratories.

Contents

Figures

Listings

Acronyms

gui	graphical user interface
IDE	Integrated Development Environment
LDRD	Laboratory Directed Research and Development
SNL	Sandia National Laboratories
xml	extended markup language

Executive Summary

This report documents the LinguisticBelief Java computer code, version 1.0. LinguisticBelief is a computer tool for the evaluation of combinations of linguistic variables. The Laboratory Directed Research and Development (LDRD) program at Sandia National Laboratories (SNL) sponsored this work. The work was performed between October 1, 2006, and January 23, 2007, at SNL in Albuquerque, New Mexico.

Using the code, linguistic variables are combined by approximate reasoning on fuzzy sets for the variables. The belief/plausibility measure of uncertainty is used to capture and propagate uncertainty for the linguistic variables. The tool is appropriate for the evaluation of variables that are difficult to express numerically. The belief/plausibility measure of uncertainty allows evaluation where there is significant epistemic uncertainty.

The mathematics of fuzzy sets, approximate reasoning, and belief/plausibility are complex. Without an automated tool, this complexity precludes their application to all but the simplest of problems. LinguisticBelief was developed to automate the use of these techniques, allowing complex problems to be easily evaluated. The code has a graphical user interface (gui) that can be used to easily create custom reasoning rule bases, apply degrees of evidence, and evaluate the rule base considering uncertainty.

Using a simple example, this report summarizes the techniques of approximate reasoning, fuzzy sets, and the belief/plausibility measure of uncertainty as implemented in LinguisticBelief. Also, the process by which a user of the tool can create a customized reasoning model is explained.

The LinguisticBelief code is being used to evaluate and rank-order potential terrorist attack scenarios. To show the use of the code to a real-world problem, this report provides summary, unclassified information on how the code is used for that application.

The belief/plausibility measure of uncertainty includes the probability measure of uncertainty as a special case, which is discussed in this report. Also, the special cases of “no uncertainty” and “total ignorance” are discussed.

An appendix summarizes the architecture and deployment of the code.

The compiled LinguisticBelief code–along with the Java runtime environment, required libraries, and a batch file for executing the code in Windows XP–all have been packaged together for deployment. The deployed package can be installed and used at no cost.

In summary, the LinguisticBelief code is a fully functional tool for creating and evaluating complex combinations of linguistic variables, including uncertainty. This report documents the use of the code.

This page intentionally left blank.

1 Introduction

LinguisticBelief is a computer tool for the evaluation of combinations of linguistic variables. Using the tool, linguistic variables are combined by approximate reasoning on fuzzy sets for the variables. The belief/plausibility measure of uncertainty is used to capture and propagate uncertainty for the linguistic variables. The tool is appropriate for the evaluation of variables that are difficult to express numerically. The belief/plausibility measure of uncertainty allows evaluation where there is significant epistemic uncertainty.

The mathematics of fuzzy sets, approximate reasoning, and belief/plausibility are complex. Without an automated tool, this complexity precludes their application to all but the simplest of problems. LinguisticBelief automates the use of these techniques, allowing complex problems to be easily evaluated. The code has a graphical user interface (gui) that can be used to easily create custom reasoning rule bases, apply degrees of evidence, and evaluate the rule base considering uncertainty.

LinguisticBelief is a 100% Java application written by the author. It was developed using the free NetBeans 5 Integrated Development Environment (IDE) from Sun Microsystems. LinguisticBelief requires “Java 5” (JSDK5) or higher as it uses features new to the language, such as generics.

LinguisticBelief uses the JFreeChart class library to create graphs of results. [Gilbert]

The compiled LinguisticBelief code–along with the Java runtime environment, required libraries, and a batch file for executing the code in Windows XP–all have been packaged together for deployment. The deployed package can be installed and used at no cost, and can be obtained by contacting the author of this report at jldarby@sandia.gov.

The techniques used in LinguisticBelief are documented in detail in a reference that summarizes the belief/plausibility measure of uncertainty, fuzzy sets, and approximate reasoning as applied to the linguistic evaluation approach used in LinguisticBelief. [Darby]

For the purposes of this report, the following summary information is provided. When we use linguistics (words) to classify events, the words have a type of uncertainty called “vagueness.” For example, yesterday was “sunny,” public confidence in the stock market is “high,” etc. Vagueness is uncertainty as to how to classify a *known* event. For example, assume we know how tall John is, but instead of saying John is 6 feet 2 inches tall we categorize John as “tall” without a precise definition of “tall.” The linguistic (word) “tall’ is vague. Vagueness can be addressed using the mathematics of fuzzy sets.

The belief/plausibility measure of uncertainty from the Dempster/Shafer Theory of Evidence is an extension of the probability measure of uncertainty that can better capture epistemic uncertainty. Belief/plausibility is a superset of probability and under certain conditions, belief and plausibility both become probability. Under other conditions, belief/plausibility become necessity/possibility, respectively. Belief/plausibility addresses a type of uncertainty called “ambiguity.” The uncertainty associated with predicting an event in the future is ambiguity.

A simple example illustrates the difference between aleatory (stochastic or "random") uncertainty and epistemic (state-of-knowledge) uncertainty, and the use of a belief/plausibility measure. Consider a fair coin, heads on one side, tails on the other, with each side equally likely. The uncertainty as to the outcome of a toss–heads or tails–is aleatory. The probability of heads is ½ and the probability of tails is ½. The uncertainty is due to the randomness of the toss. Suppose, however, that I do not know the coin is fair; the coin could be biased to come up heads, or the coin could even be two-tailed. Now that I have epistemic uncertainty, my state of knowledge is insufficient to assign a probability to heads or tails; all I can say is the likelihood of heads (or tails) is somewhere between 0 and 1. To consider epistemic uncertainty as well as aleatory uncertainty, belief/plausibility can be used as the measure of uncertainty. With total ignorance about the coin, the belief that the toss will be heads is 0 and the plausibility that the toss will be heads is 1; similarly, the belief that the toss will be tails is 0 and the plausibility that the toss will be tails is 1. Belief/plausibility form an interval that can be interpreted as giving the lower and upper bound of probability. If I have enough information, both belief and plausibility reduce to a single value, probability. Figure 1-1 illustrates this concept. Epistemic uncertainty can be reduced with more information. If I toss the coin a few times and a heads and a tails occur, I know the coin is two-sided; with more tosses I can evaluate the fairness of the coin. Aleatory uncertainty cannot be reduced with more information.

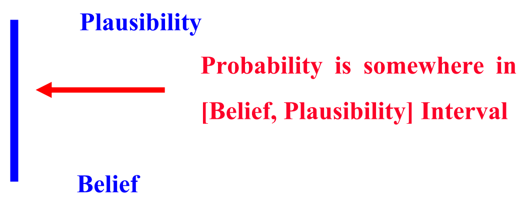


Figure 1-1. Belief/Plausibility as Bounds on Probability

Approximate reasoning is a rule base for combining fuzzy sets for different linguistic variables. For example, "Bad" "Health" and "Poor" "Wealth" cause "Not so Good" "Quality of Life."

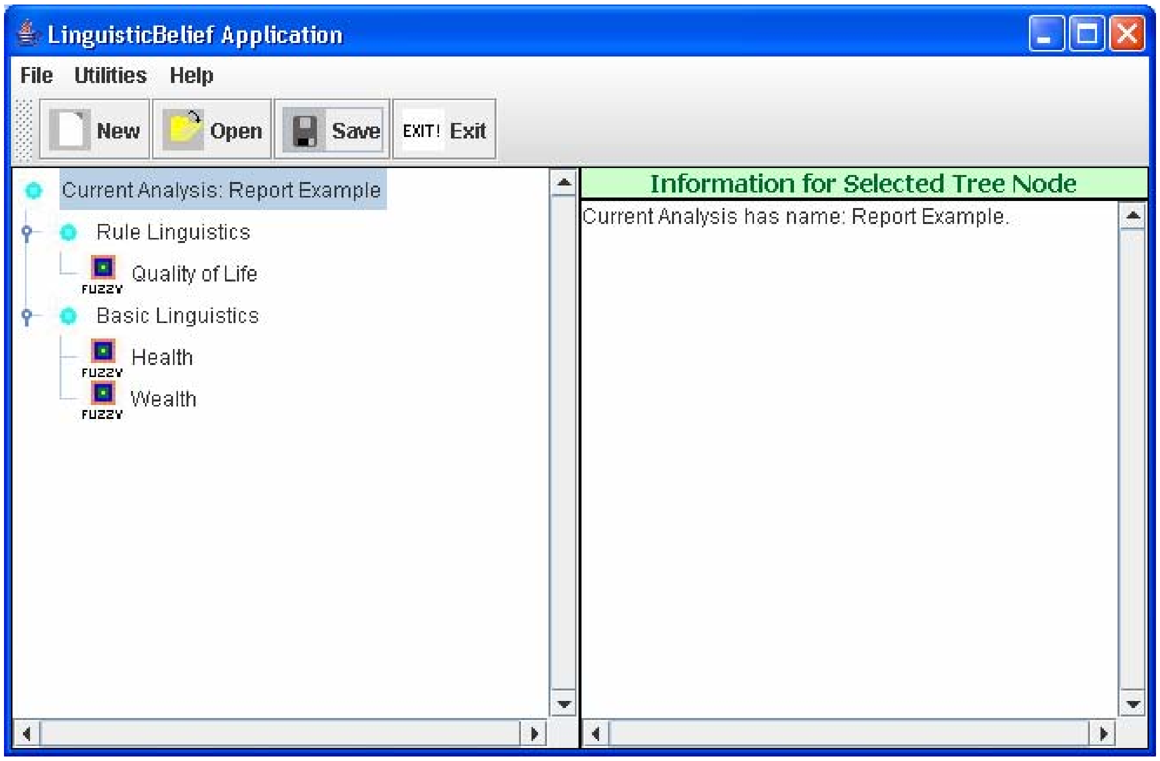
The LinguisticBelief code uses an approximate reasoning rule base to evaluate combinations of linguistic variables expressed as fuzzy sets, considering uncertainty using the belief/plausibility measure.

1.1 Overview of LinguisticBelief

An overview of LinguisticBelief is summarized in this section using a simple example. Section 2.3 shows how this example is created in LinguisticBelief.

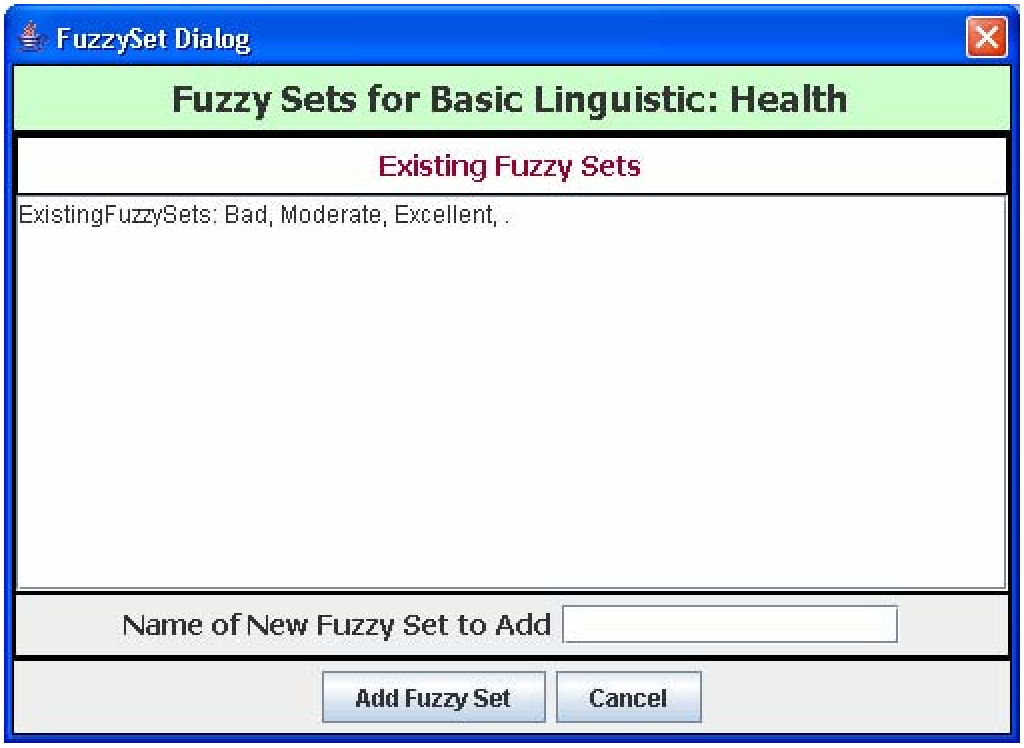
Assume we wish to reason on "Quality of Life" based on "Health" and "Wealth." "Health" and "Wealth" are basic linguistic variables, and "Quality of Life" is a rule-based linguistic variable that it is evaluated by a set of rules for combining "Health" and "Wealth." Figure 1-2 shows these linguistics entered into LinguisticBelief.

For “Health” we chose to use the fuzzy sets “Bad,” “Moderate,” and “Excellent.” For “Wealth” we use the fuzzy sets “Poor,” “Middle Class,” and “Rich.” For “Quality of Life” we use the fuzzy sets “Good” and “Not so Good.” Figure 1-3 shows the fuzzy sets for “Health” entered into LinguisticBelief.



Current Analysis has name: Report Example.

Figure 1-2. Linguistics for Reasoning on “Quality of Life”



Fuzzy Sets for Basic Linguistic: Health

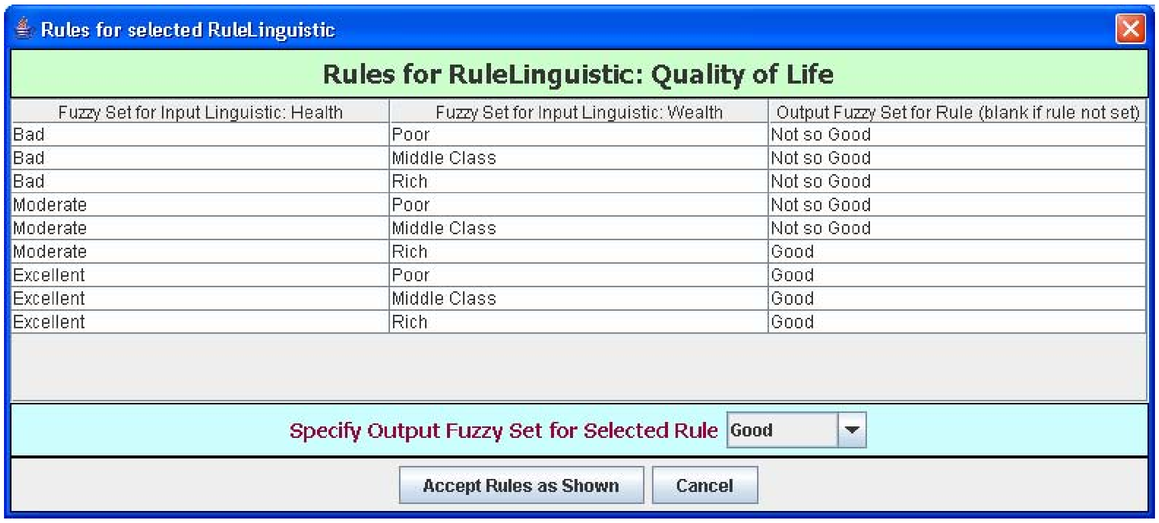
Existing Fuzzy Sets

ExistingFuzzySets: Bad, Moderate, Excellent, .

Name of New Fuzzy Set to Add

Figure 1-3. Fuzzy Sets for "Health"

Figure 1-4 shows the user-created approximate reasoning rule base used to evaluate "Quality of Life" based on "Health" and "Wealth."[1] The rule base specifies the mapping of combinations of input fuzzy sets to output fuzzy sets. For example, the rule base in Figure 1-4 indicates that "Bad" "Health" and ""Poor" "Wealth" result in "Not so Good" "Quality of Life."



Rules for RuleLinguistic: Quality of Life

Fuzzy Set for Input Linguistic: Health	Fuzzy Set for Input Linguistic: Wealth	Output Fuzzy Set for Rule (blank if rule not set)
Bad	Poor	Not so Good
Bad	Middle Class	Not so Good
Bad	Rich	Not so Good
Moderate	Poor	Not so Good
Moderate	Middle Class	Not so Good
Moderate	Rich	Good
Excellent	Poor	Good
Excellent	Middle Class	Good
Excellent	Rich	Good

Specify Output Fuzzy Set for Selected Rule: Good

Figure 1-4. Approximate Reasoning Rule Base for "Quality of Life"

[1] The variables are assumed to be non-interacting. [Darby].

This model can be used to reason on Quality of Life for *any* individual.

The Quality of Life for a *specific* individual is evaluated by assigning degrees of evidence to collections of fuzzy sets for each basic linguistic variable. Assume we have the following focal elements (evidence over families of fuzzy sets) for the "Health" of a specific individual named "John":

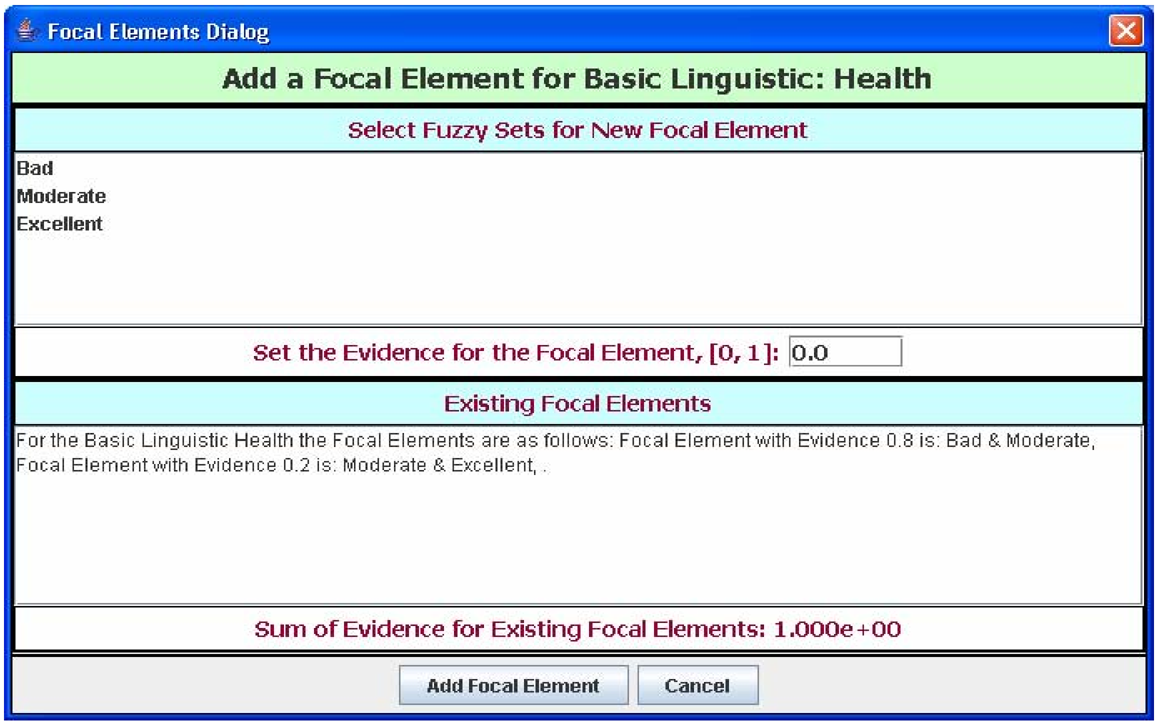
> 0.8 to {"Bad," "Moderate"}, and
> 0.2 to {"Moderate," "Excellent"}.[2]

Assume we have the following focal elements for the "Wealth" of "John":

> 0.3 to {"Middle Class"}, and
> 0.7 to {"Poor," "Middle Class"}.

Figure 1-5 shows the focal elements for the "Health" of "John" entered into LinguisticBelief; similarly, the focal elements for the "Wealth" of "John" are entered into the code.

After focal elements for all basic linguistic variables are entered, the belief/plausibility for any linguistic variable (basic or rule-based) can be calculated. Figure 1-6 shows the belief/plausibility for "Wealth" for "John," and Figure 1-7 shows the belief/plausibility for "Quality of Life" for "John."


Focal Elements Dialog

Add a Focal Element for Basic Linguistic: Health

Select Fuzzy Sets for New Focal Element

Bad
Moderate
Excellent

Set the Evidence for the Focal Element, [0, 1]: 0.0

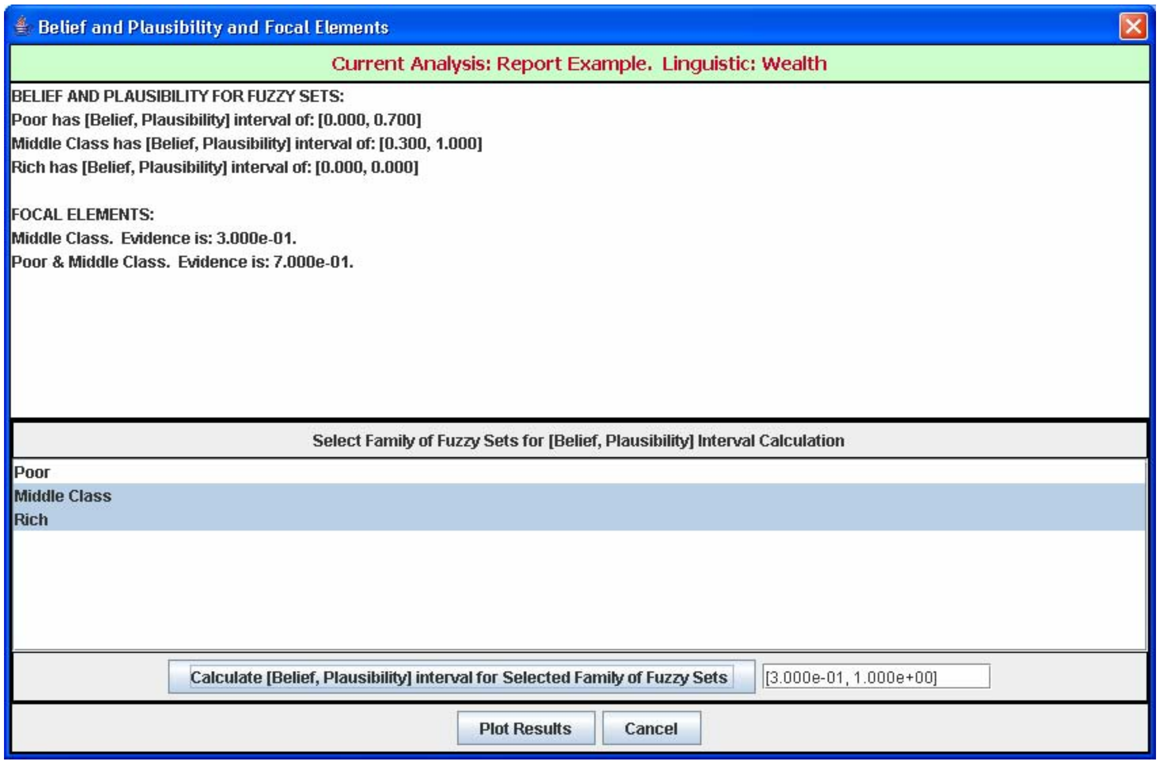
Existing Focal Elements

For the Basic Linguistic Health the Focal Elements are as follows: Focal Element with Evidence 0.8 is: Bad & Moderate, Focal Element with Evidence 0.2 is: Moderate & Excellent, .

Sum of Evidence for Existing Focal Elements: 1.000e+00

Figure 1-5. Focal Elements for "Health" of "John"

2 As discussed in a reference, the fuzziness of the sets is considered in the assignment of degrees of evidence. [Darby] For linguistic convolution in the rule base, the sets are treated as crisp. Both belief and plausibility are probability if all the focal elements are singletons; see Section 4.1 of this report.

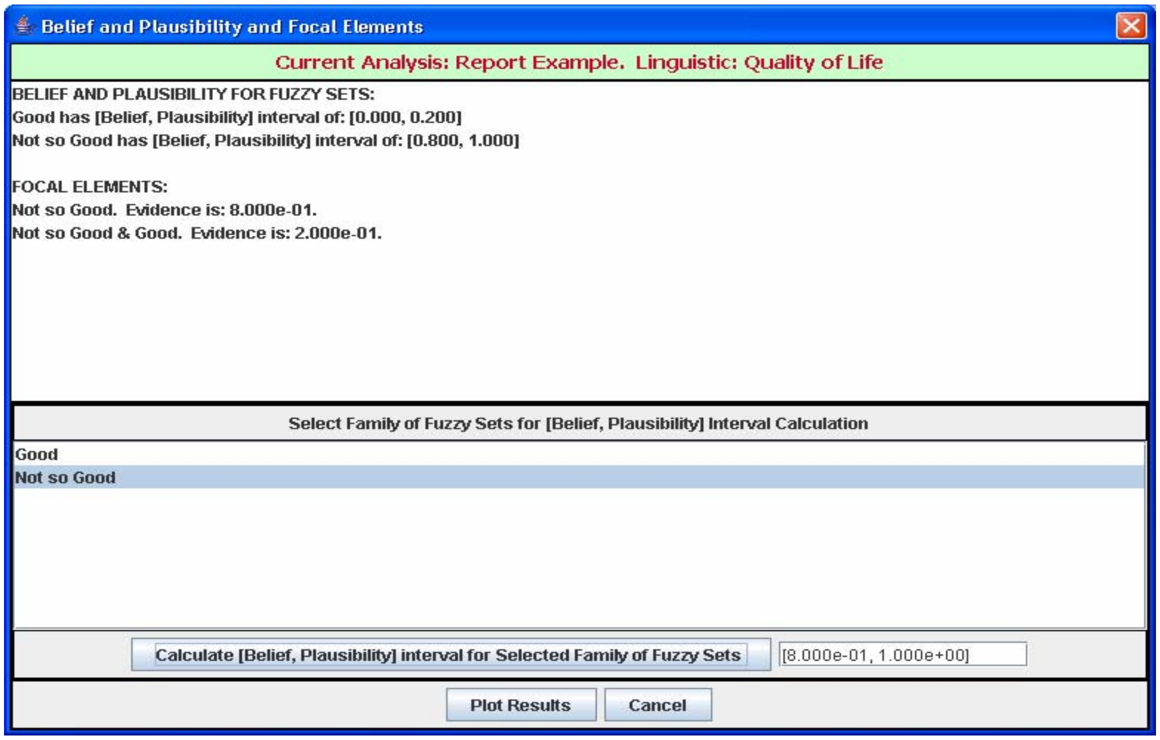

Belief and Plausibility and Focal Elements
Current Analysis: Report Example. Linguistic: Wealth

```
BELIEF AND PLAUSIBILITY FOR FUZZY SETS:
Poor has [Belief, Plausibility] interval of: [0.000, 0.700]
Middle Class has [Belief, Plausibility] interval of: [0.300, 1.000]
Rich has [Belief, Plausibility] interval of: [0.000, 0.000]

FOCAL ELEMENTS:
Middle Class.  Evidence is: 3.000e-01.
Poor & Middle Class.  Evidence is: 7.000e-01.
```

Select Family of Fuzzy Sets for [Belief, Plausibility] Interval Calculation: Poor, Middle Class (selected), Rich (selected)

Calculate [Belief, Plausibility] interval for Selected Family of Fuzzy Sets: [3.000e-01, 1.000e+00]

Figure 1-6. Belief/Plausibility for “Wealth” for “John”


Belief and Plausibility and Focal Elements
Current Analysis: Report Example. Linguistic: Quality of Life

```
BELIEF AND PLAUSIBILITY FOR FUZZY SETS:
Good has [Belief, Plausibility] interval of: [0.000, 0.200]
Not so Good has [Belief, Plausibility] interval of: [0.800, 1.000]

FOCAL ELEMENTS:
Not so Good.  Evidence is: 8.000e-01.
Not so Good & Good.  Evidence is: 2.000e-01.
```

Select Family of Fuzzy Sets for [Belief, Plausibility] Interval Calculation: Good, Not so Good (selected)

Calculate [Belief, Plausibility] interval for Selected Family of Fuzzy Sets: [8.000e-01, 1.000e+00]

Figure 1-7. Belief/Plausibility for “Quality of Life” for “John”

The belief/plausibility results from the code provide three categories of information for the specific linguistic variable of interest. First, the [belief, plausibility] interval for each of the fuzzy sets is given; for example, Figure 1-6 shows the following for "Wealth" for "John:"

BELIEF AND PLAUSIBILITY FOR FUZZY SETS:
Poor has [Belief, Plausibility] Interval of: [0.000, 0.700]
Middle Class has [Belief, Plausibility] Interval of: [0.300, 1.000]
Rich has [Belief, Plausibility] Interval of: [0.000, 0.000]

Second, the focal elements for the linguistic variable are provided; for example, Figure 1-6 shows the following focal elements for "Wealth" for "John:"

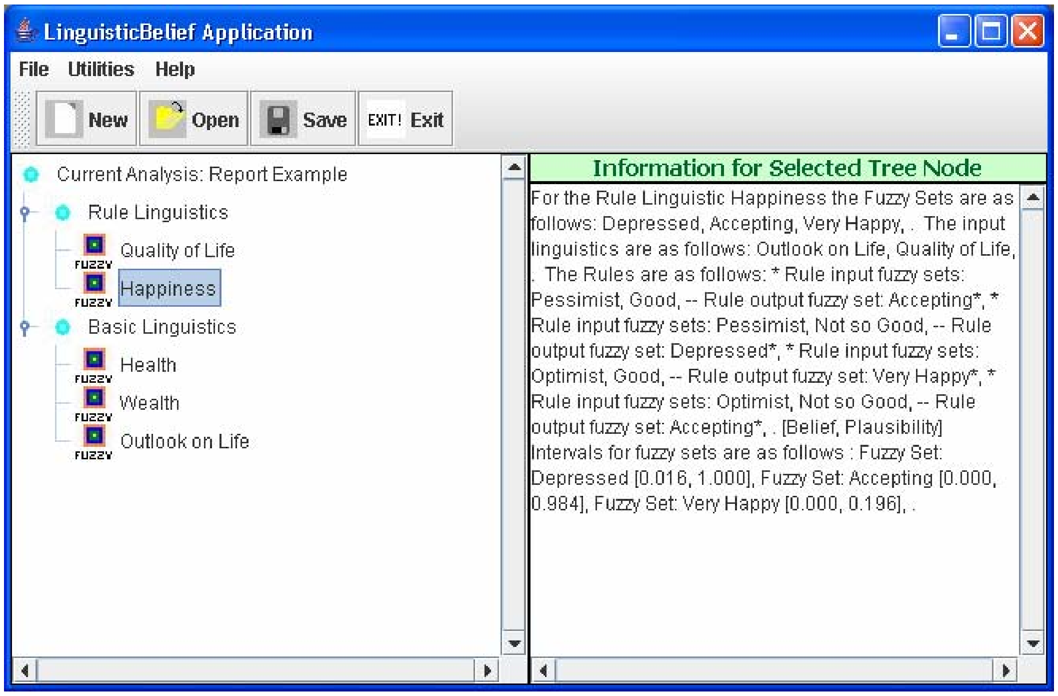
FOCAL ELEMENTS:
Middle Class. Evidence is 3.000e-01.
Poor & Middle Class. Evidence is 7.000e-01.

Third, the [belief, plausibility] interval for any family of fuzzy sets can be calculated; for example, Figure 1-6 shows the [belief, plausibility] for the family {"Middle Class," "Rich"} for the "Wealth" for "John" as:

[3.000e-01, 1.000e+00].

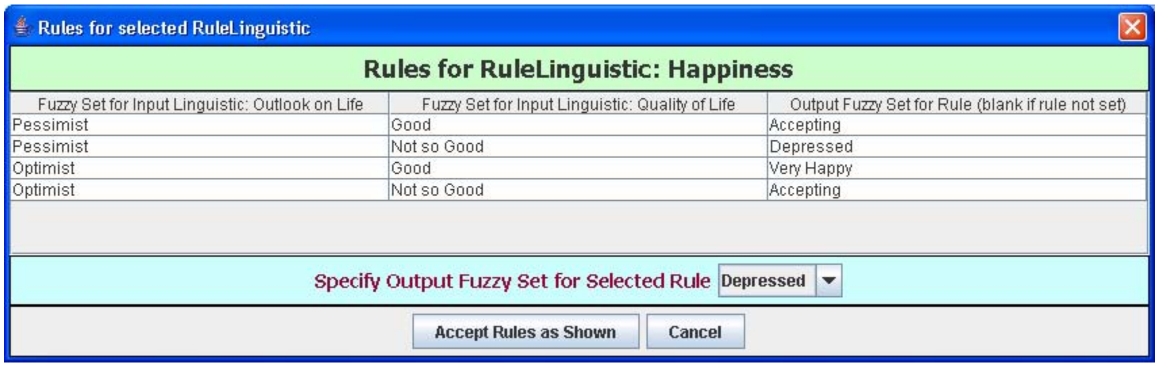
This information answers such questions as: "What is the belief/plausibility that "John" is wealthier than 'Poor'?" (This is the belief/plausibility for {"Middle Class," "Rich"} which is 0.3/1.0.)

Assume we expand the model to reason on "Happiness," another rule-based linguistic variable, as a combination of "Quality of Life" and a new basic linguistic variable "Outlook on Life." The fuzzy sets for "Outlook on Life" are specified as: "Pessimist" and "Optimist." The fuzzy sets for "Happiness" are chosen to be: "Depressed," "Accepting," and "Very Happy." Figure 1-8 shows the expanded model in LinguisticBelief, and Figure 1-9 shows the user-specified approximate reasoning rule base for "Happiness" based on "Quality of Life" and "Outlook on Life."



For the Rule Linguistic Happiness the Fuzzy Sets are as follows: Depressed, Accepting, Very Happy, . The input linguistics are as follows: Outlook on Life, Quality of Life, . The Rules are as follows: * Rule input fuzzy sets: Pessimist, Good, -- Rule output fuzzy set: Accepting*, * Rule input fuzzy sets: Pessimist, Not so Good, -- Rule output fuzzy set: Depressed*, * Rule input fuzzy sets: Optimist, Good, -- Rule output fuzzy set: Very Happy*, * Rule input fuzzy sets: Optimist, Not so Good, -- Rule output fuzzy set: Accepting*, . [Belief, Plausibility] Intervals for fuzzy sets are as follows : Fuzzy Set: Depressed [0.016, 1.000], Fuzzy Set: Accepting [0.000, 0.984], Fuzzy Set: Very Happy [0.000, 0.196], .

Figure 1-8. Model Expanded to include "Happiness"


Rules for selected RuleLinguistic
Rules for RuleLinguistic: Happiness

Fuzzy Set for Input Linguistic: Outlook on Life	Fuzzy Set for Input Linguistic: Quality of Life	Output Fuzzy Set for Rule (blank if rule not set)
Pessimist	Good	Accepting
Pessimist	Not so Good	Depressed
Optimist	Good	Very Happy
Optimist	Not so Good	Accepting

Specify Output Fuzzy Set for Selected Rule: Depressed

Accept Rules as Shown
Cancel

Figure 1-9. Approximate Reasoning Rule Base for "Happiness"

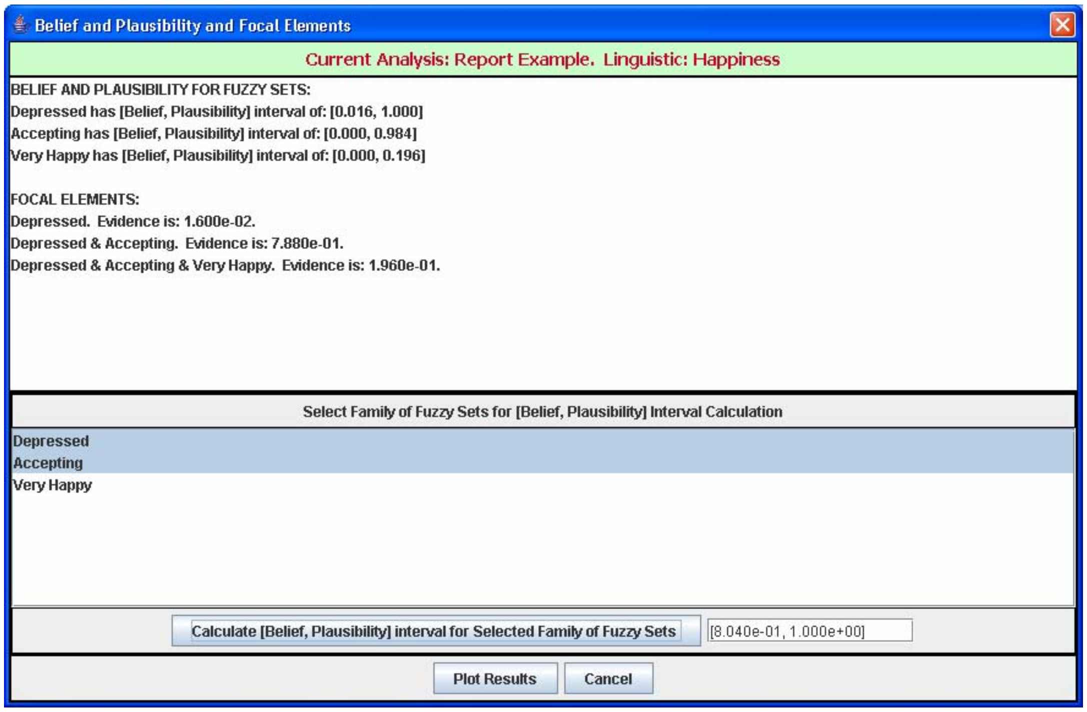
Note that "Happiness," a rule-based linguistic, is a combination of "Quality of Life," another rule-based linguistic, and "Outlook on Life," a basic linguistic. The LingusticBelief code allows a rule-based linguistic to be a combination of linguistics, basic or rule-based. Circular logic is not allowed; that is, no rule-based linguistic can have itself as an input linguistic. The code recursively checks any proposed input linguistic that is itself a rule-based linguistic for circular logic.

Assume the following focal elements for "Outlook on Life" for "John:"

{"Pessimist"} with evidence 0.02, and

{"Pessimist," "Optimist"} with evidence 0.98.

Figure 1-10 shows the belief/plausibility results for "Happiness" for "John."



Current Analysis: Report Example. Linguistic: Happiness

```
BELIEF AND PLAUSIBILITY FOR FUZZY SETS:
Depressed has [Belief, Plausibility] interval of: [0.016, 1.000]
Accepting has [Belief, Plausibility] interval of: [0.000, 0.984]
Very Happy has [Belief, Plausibility] interval of: [0.000, 0.196]

FOCAL ELEMENTS:
Depressed. Evidence is: 1.600e-02.
Depressed & Accepting. Evidence is: 7.880e-01.
Depressed & Accepting & Very Happy. Evidence is: 1.960e-01.
```

Select Family of Fuzzy Sets for [Belief, Plausibility] Interval Calculation

Depressed
Accepting
Very Happy

Calculate [Belief, Plausibility] interval for Selected Family of Fuzzy Sets: [8.040e-01, 1.000e+00]

Figure 1-10. Belief/Plausibility for "Happiness" for "John"

Using the **Plot Results** button in Figure 1-10, the graphical results in Figure 1-11 are produced.

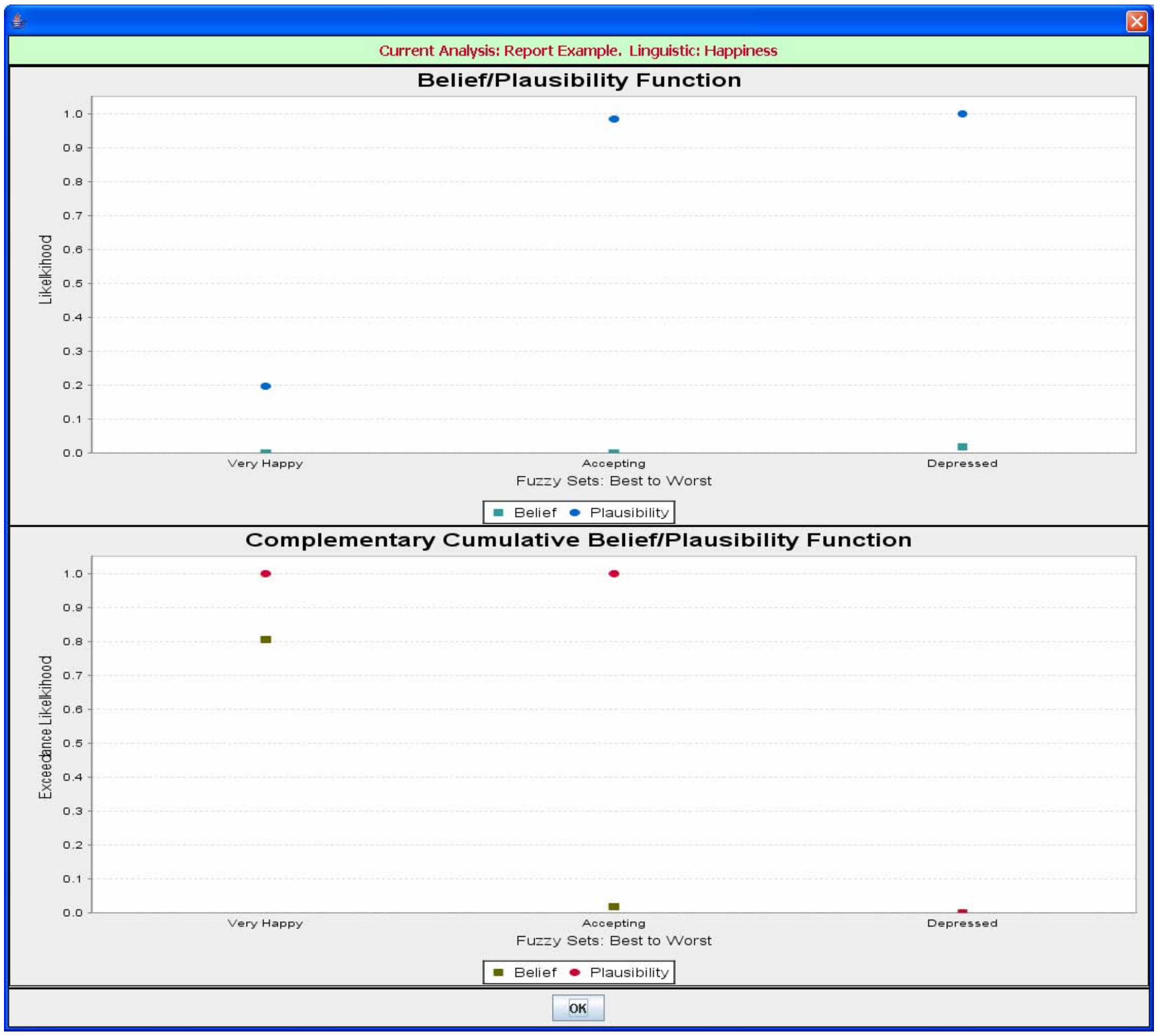


Figure 1-11. Graphical Results for "Happiness" for "John"

The top graph in Figure 1-11 is the "likelihood" of each fuzzy set, where likelihood is a belief/plausibility interval. The fuzzy sets for "Happiness" in Figure 1-11 are ordered from most happy to least happy as: "Very Happy," "Accepting," and "Depressed." Using this ordering, the likelihood of **exceedance** for each of these fuzzy set can be calculated. The lower graph in Figure 1-11 shows a high belief/plausibility of exceedance for "Very Happy" of 0.80/1.0.[3] That is, it is highly likely that "John" is not "Very Happy."

The top graph in Figure 1-11 is the belief/plausibility distribution for "Happiness" for "John." It is similar to the probability density distribution for a discrete random (numerical) variable,

[3] The likelihood of *exceedance* for a fuzzy set is the likelihood that the variable of interest is *worse* than the indicated fuzzy set. That is, in the lower graph of Figure 1-11, the belief/plausibility of exceedance for "Very Happy" is the belief/plausibility for "Happiness" being worse than "Very Happy." The belief/plausibility of exceedance for "Very Happy" is calculated as the belief/plausibility for {"Accepting, "Depressed"}.

except the values are fuzzy sets, not numbers, and the measure of likelihood is a belief/plausibility interval, not a point probability.[4] The bottom graph in Figure 1-11 is the complementary cumulative belief/plausibility distribution for "Happiness" for "John." It is similar to the complementary cumulative probability distribution for a discrete random (numerical) variable, except the values are fuzzy sets, not numbers, and the measure of likelihood is a belief/plausibility interval, not a point probability.

An individual other than "John" can be evaluated by modifying the focal elements in the model to reflect the appropriate evidence. For example, suppose the information available for "Janet" yields the following focal elements. For "Health:"

0.3 to {"Moderate"}, and
0.7 to {"Moderate," "Excellent"}.

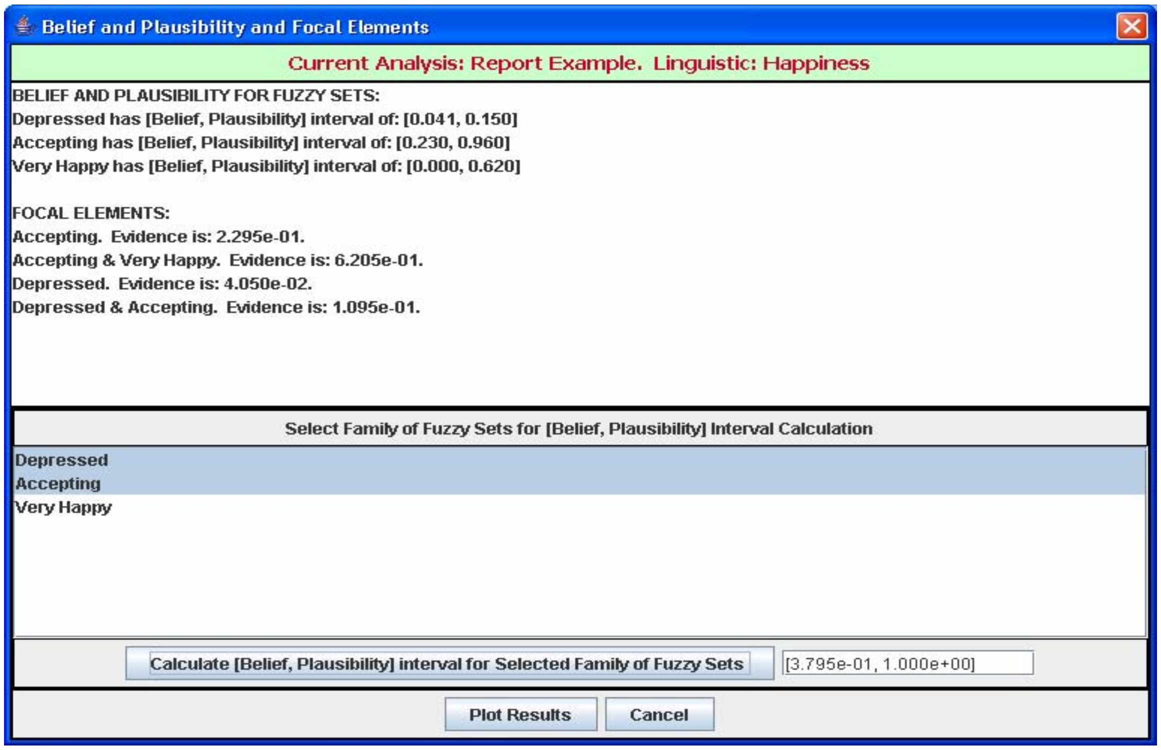
For "Wealth":

0.9 to {"Middle Class"}, and
0.1 to {"Middle Class," "Rich"}.

For "Outlook on Life":

{"Optimist"} with evidence 0.85, and
{"Pessimist,"} with evidence 0.15.

Figures 1-12 and 1-13 show the "Happiness" for "Janet."


Belief and Plausibility and Focal Elements

Current Analysis: Report Example. Linguistic: Happiness

BELIEF AND PLAUSIBILITY FOR FUZZY SETS:
Depressed has [Belief, Plausibility] interval of: [0.041, 0.150]
Accepting has [Belief, Plausibility] interval of: [0.230, 0.960]
Very Happy has [Belief, Plausibility] interval of: [0.000, 0.620]

FOCAL ELEMENTS:
Accepting. Evidence is: 2.295e-01.
Accepting & Very Happy. Evidence is: 6.205e-01.
Depressed. Evidence is: 4.050e-02.
Depressed & Accepting. Evidence is: 1.095e-01.

Select Family of Fuzzy Sets for [Belief, Plausibility] Interval Calculation

Depressed
Accepting
Very Happy

Calculate [Belief, Plausibility] interval for Selected Family of Fuzzy Sets: [3.795e-01, 1.000e+00]

Figure 1-12. Belief/Plausibility Distribution for "Happiness" for "Janet"

[4] If all the focal elements are singletons, each belief/plausibility interval degenerates into a point that is the probability. In this situation the results are probability distributions over fuzzy sets. Section 4.1 provides an example.

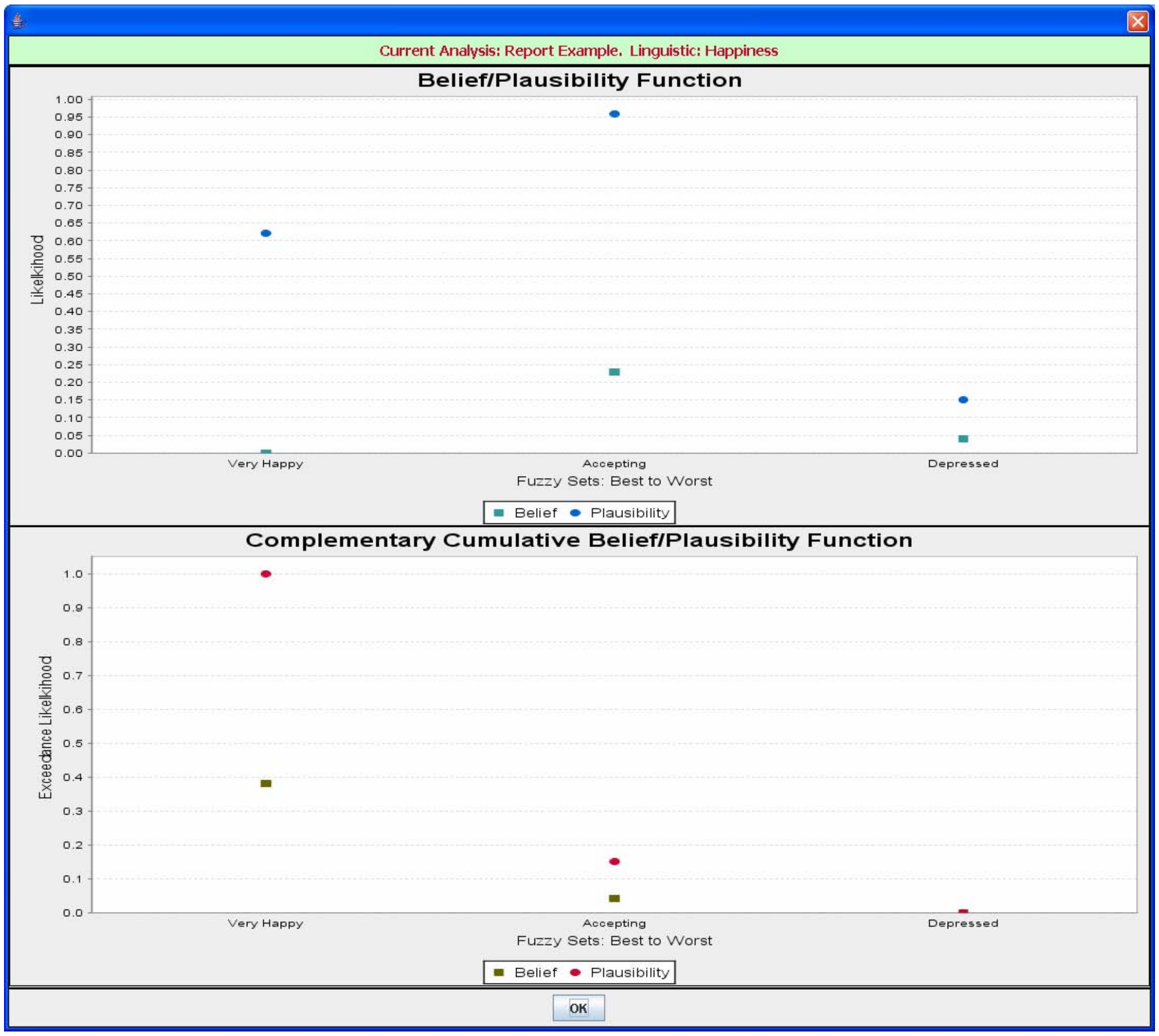

Current Analysis: Report Example. Linguistic: Happiness

Figure 1-13. Graphical Results for "Happiness" for "Janet"

It is likely that "Janet" is "Very Happy" or "Accepting" since the belief/plausibility for exceedance of "Accepting" is only 0.04/0.15.

Using plausibility, the results imply that "John" is less happy than "Janet." The plausibility that "John" is "Depressed" is 1.0, and the plausibility that "Janet" is "Depressed" is 0.15.[5]

5 Section 3.2 provides an algorithm for ranking results using both plausibility and belief, and this example is evaluated using that algorithm in Section 3.2.

2 Using LinguisticBelief

Section 2.1 summarizes the top-level structure of the LinguisticBelief gui and Section 2.3 shows how to create an analysis in LinguisticBelief.

2.1 Structure of LinguisticBelief Graphical User Interface

Figure 2-1 is the frame (JFrame) for LinguisticBelief for a New analysis. A New analysis is an empty analysis, and is created by starting the application or by selecting either the **New** button or the **File|New** menu item.

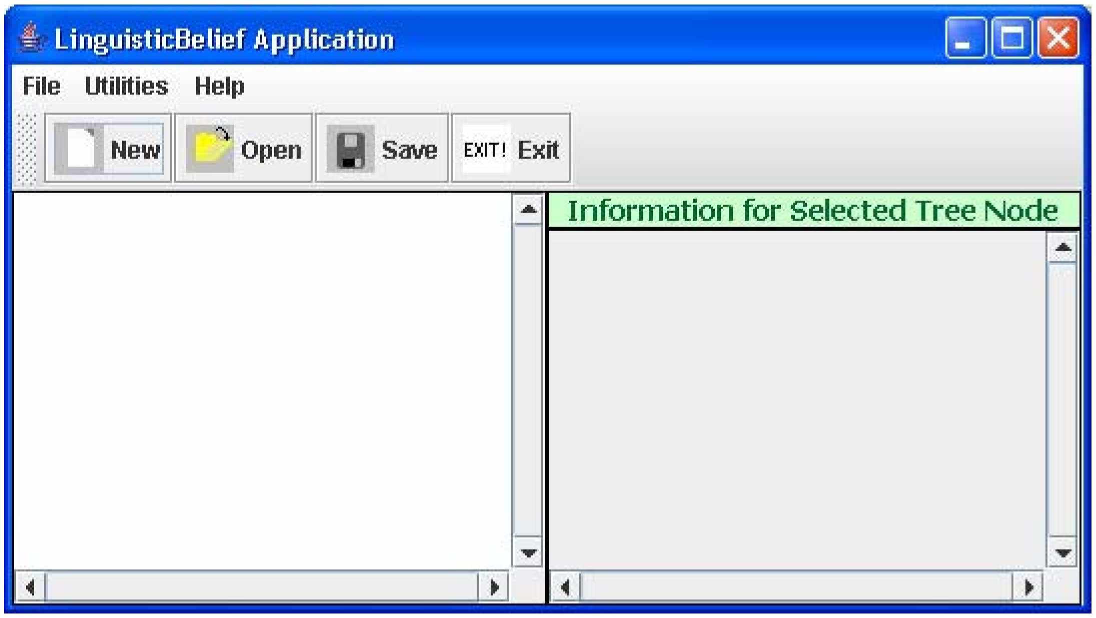
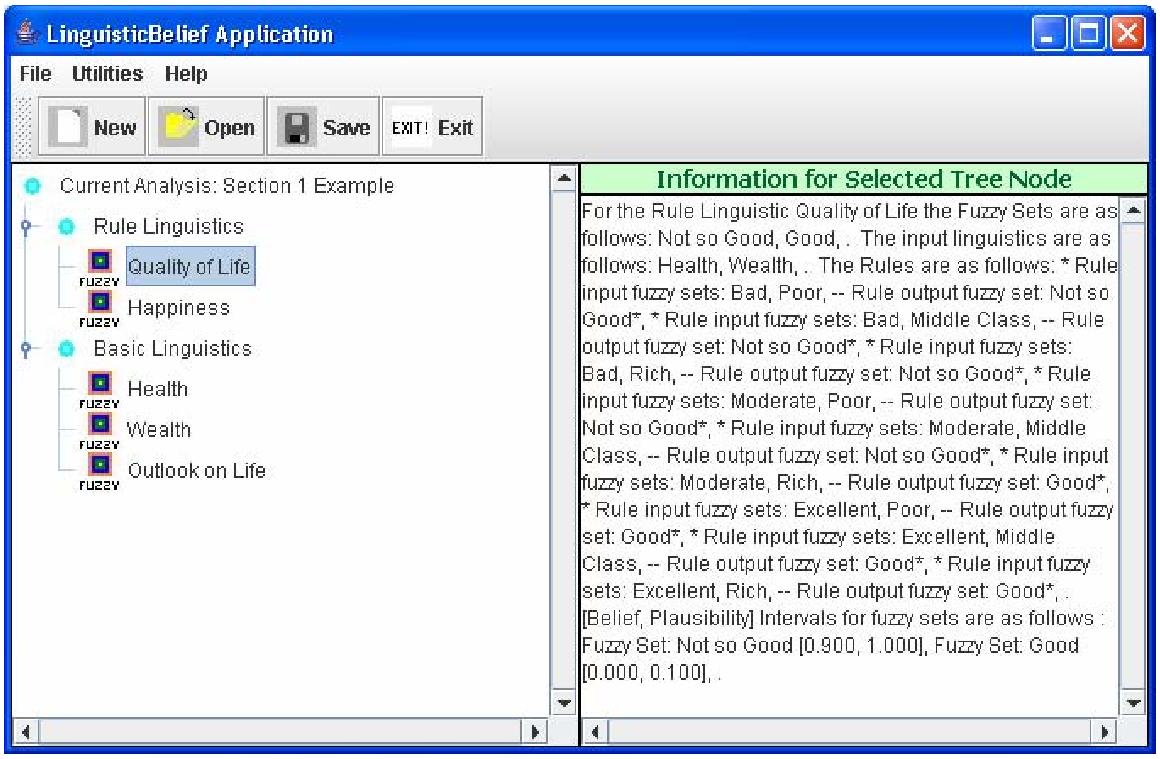

LinguisticBelief Application
File
Utilities
Help
New
Open
Save
EXIT!
Exit
Information for Selected Tree Node

Figure 2-1. A New Analysis in LinguisticBelief

2.2 Analysis Structure

Figure 2-2 shows the frame containing the analysis discussed in Section 1. The frame has:

- a menu bar (**File, Utilities, Help**)
- a button bar (**New, Open, Save, Exit**)
- two panels (JPanels):
    - The **left panel** displays the linguistics for the current analysis in a tree form (JTree),
    - the **right panel** displays information on the state of a linguistic variable selected in the tree in the left panel.



For the Rule Linguistic Quality of Life the Fuzzy Sets are as follows: Not so Good, Good, . The input linguistics are as follows: Health, Wealth, . The Rules are as follows: * Rule input fuzzy sets: Bad, Poor, -- Rule output fuzzy set: Not so Good*, * Rule input fuzzy sets: Bad, Middle Class, -- Rule output fuzzy set: Not so Good*, * Rule input fuzzy sets: Bad, Rich, -- Rule output fuzzy set: Not so Good*, * Rule input fuzzy sets: Moderate, Poor, -- Rule output fuzzy set: Not so Good*, * Rule input fuzzy sets: Moderate, Middle Class, -- Rule output fuzzy set: Not so Good*, * Rule input fuzzy sets: Moderate, Rich, -- Rule output fuzzy set: Good*, * Rule input fuzzy sets: Excellent, Poor, -- Rule output fuzzy set: Good*, * Rule input fuzzy sets: Excellent, Middle Class, -- Rule output fuzzy set: Good*, * Rule input fuzzy sets: Excellent, Rich, -- Rule output fuzzy set: Good*, . [Belief, Plausibility] Intervals for fuzzy sets are as follows : Fuzzy Set: Not so Good [0.900, 1.000], Fuzzy Set: Good [0.000, 0.100], .

Figure 2-2. An Existing Analysis in LinguisticBelief

In Figure 2-2, the “Quality of Life” linguistic variable is selected in the tree in the left panel. The information on the current state of “Quality of Life” is provided in the right panel.

2.2.1 Commonly Used Options

The following commonly used options are provided on the button bar:

- **New**—Creates an empty analysis
- **Open**—Loads an old analysis previously saved in xml format
- **Save**—Saves the current analysis in extended markup language (xml) format[6]
- **Exit**—Closes the application

All of these options, and other less frequently used options (**Save As** and **Delete**), are available under the **File** menu as shown in Figure 2-3. Other options include:

- **Save As**—Saves the current analysis in xml format under a different user-specified name, allowing further modifications to be made to the newly-named analysis without affecting the state of the prior analysis.
- **Delete**—Allows the user to remove an analysis from the collection of saved xml analyses

[6] Analyses can also be saved and loaded in Java serialized (binary) form using the Utilities menu as subsequently discussed. Typically, analyses are saved and opened in the xml format.

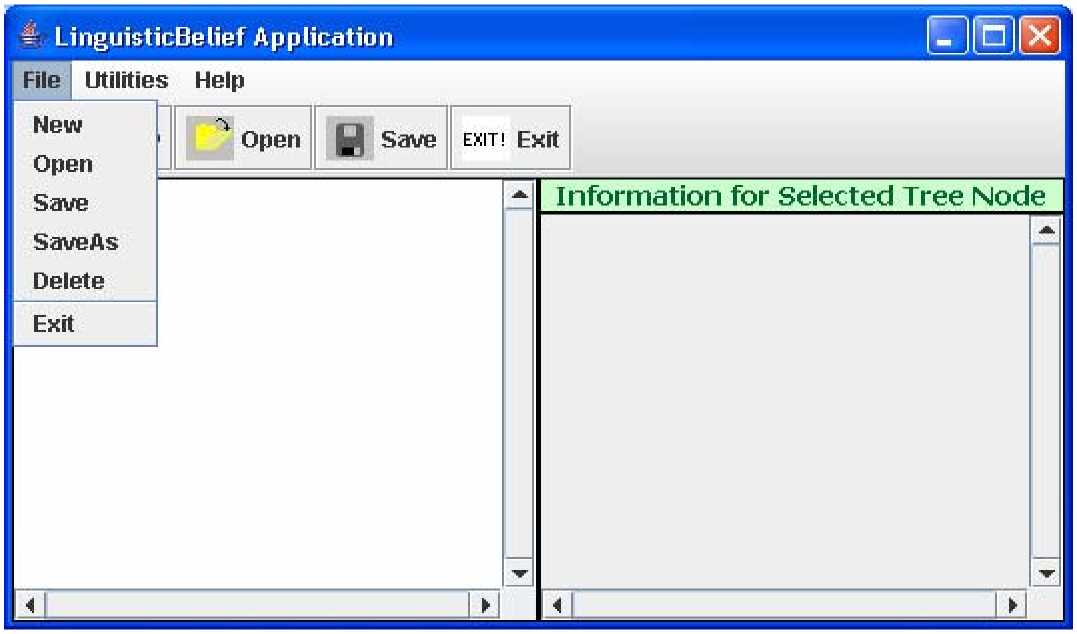

LinguisticBelief Application
File
Utilities
Help
New
Open
Save
SaveAs
Delete
Exit
Open
Save
EXIT! Exit
Information for Selected Tree Node

Figure 2-3. Options under the File Menu

2.2.2 Help Menu Options

The Help menu provides the options shown in Figure 2-4.

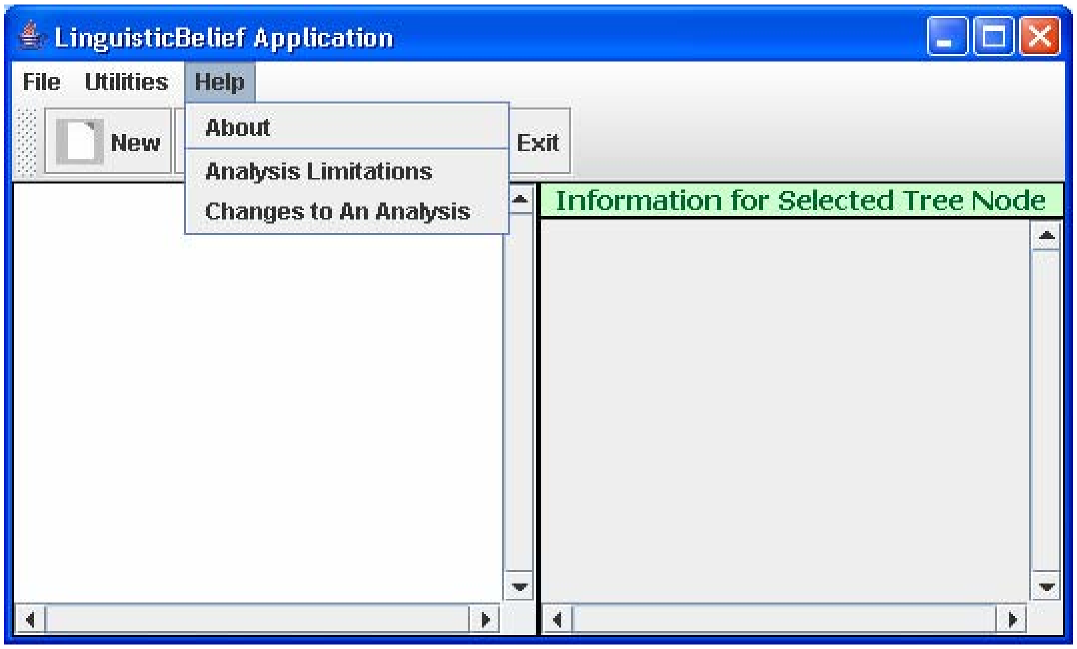
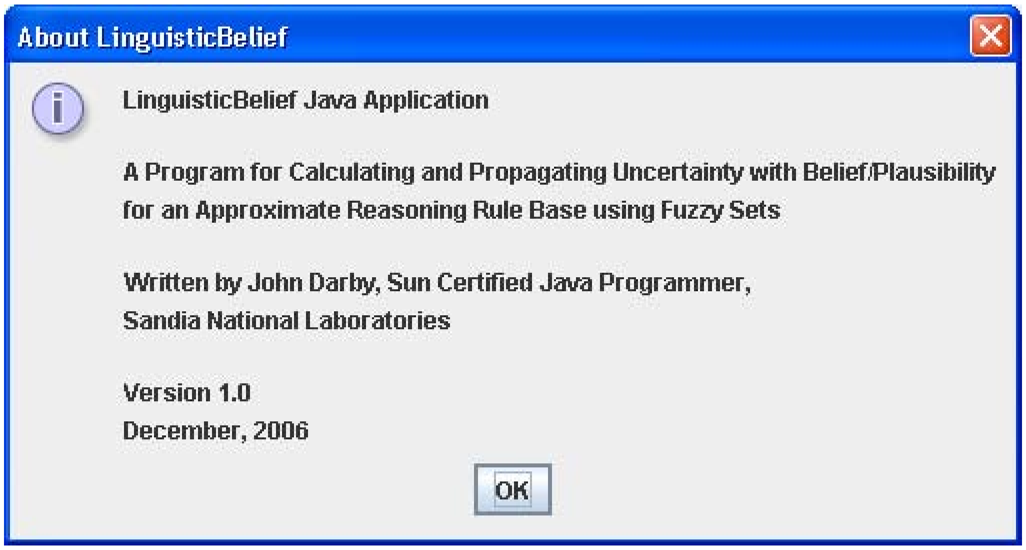


Figure 2-4. Options under the Help Menu

The information provided by each of these Help options is shown in Figures 2-5 through 2-7.


About LinguisticBelief

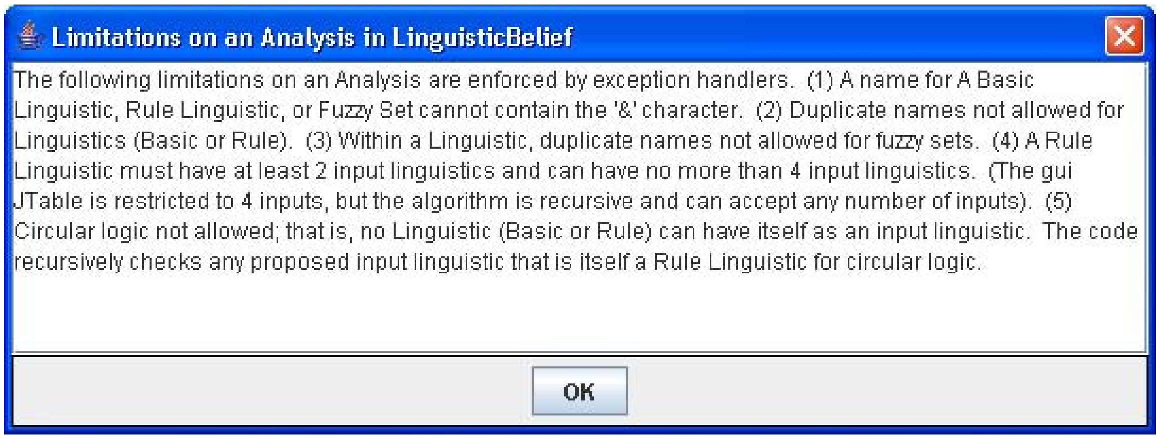
LinguisticBelief Java Application

A Program for Calculating and Propagating Uncertainty with Belief/Plausibility for an Approximate Reasoning Rule Base using Fuzzy Sets

Written by John Darby, Sun Certified Java Programmer, Sandia National Laboratories

Version 1.0
December, 2006

Figure 2-5. The Menu Item Help|About

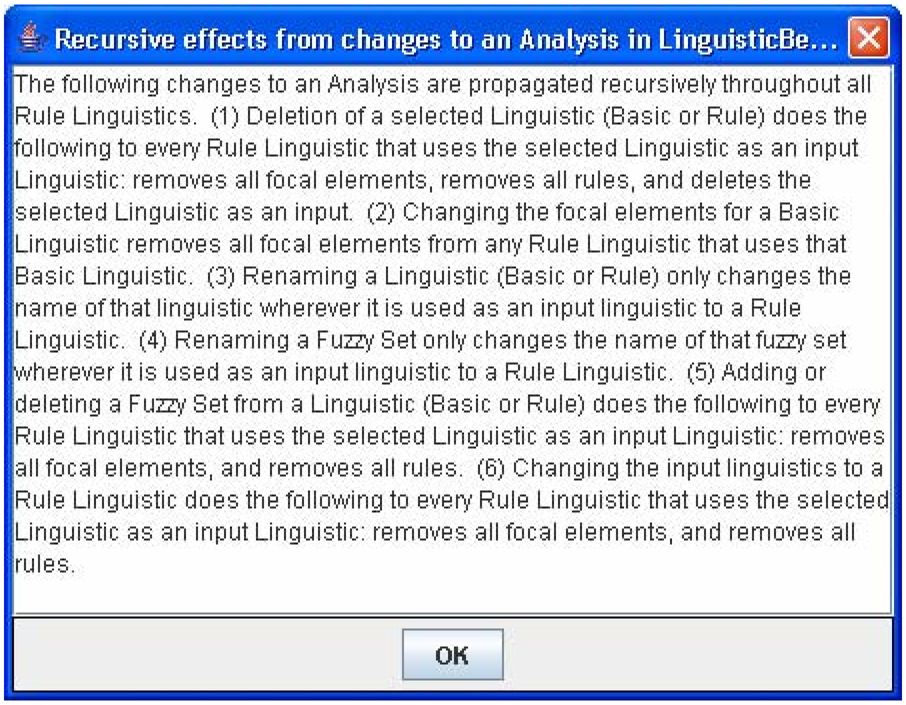

Limitations on an Analysis in LinguisticBelief

The following limitations on an Analysis are enforced by exception handlers. (1) A name for A Basic Linguistic, Rule Linguistic, or Fuzzy Set cannot contain the '&' character. (2) Duplicate names not allowed for Linguistics (Basic or Rule). (3) Within a Linguistic, duplicate names not allowed for fuzzy sets. (4) A Rule Linguistic must have at least 2 input linguistics and can have no more than 4 input linguistics. (The gui JTable is restricted to 4 inputs, but the algorithm is recursive and can accept any number of inputs). (5) Circular logic not allowed; that is, no Linguistic (Basic or Rule) can have itself as an input linguistic. The code recursively checks any proposed input linguistic that is itself a Rule Linguistic for circular logic.

Figure 2-6. The Menu Item Help|Analysis Limitations

The "**Help|Analysis Limitations**" dialog shown in Figure 2-6 summarizes the limitations enforced by exception handlers in the code. For example, each linguistic variable must have a unique name, and each rule-based linguistic must have at least two and no more than four input linguistics.[7]

[7] The limitation of four inputs in the gui is a practical one, based on the difficulty of the user assigning rules with combinations of more than four inputs. Any linguistic variable requiring more than four inputs can be modified to have no more than four inputs, the inputs themselves having inputs as necessary to reflect the desired reasoning rule base. The algorithm that evaluates inputs is *not* restricted to four inputs; it is recursive and can accept any number of inputs as discussed in Appendix A. The four input limitation is on the gui. In the future, rule linguistics with more than four inputs may be input from other tools and can be handled by the algorithm in LinguisticBelief.


Recursive effects from changes to an Analysis in LinguisticBe...
OK

The following changes to an Analysis are propagated recursively throughout all Rule Linguistics. (1) Deletion of a selected Linguistic (Basic or Rule) does the following to every Rule Linguistic that uses the selected Linguistic as an input Linguistic: removes all focal elements, removes all rules, and deletes the selected Linguistic as an input. (2) Changing the focal elements for a Basic Linguistic removes all focal elements from any Rule Linguistic that uses that Basic Linguistic. (3) Renaming a Linguistic (Basic or Rule) only changes the name of that linguistic wherever it is used as an input linguistic to a Rule Linguistic. (4) Renaming a Fuzzy Set only changes the name of that fuzzy set wherever it is used as an input linguistic to a Rule Linguistic. (5) Adding or deleting a Fuzzy Set from a Linguistic (Basic or Rule) does the following to every Rule Linguistic that uses the selected Linguistic as an input Linguistic: removes all focal elements, and removes all rules. (6) Changing the input linguistics to a Rule Linguistic does the following to every Rule Linguistic that uses the selected Linguistic as an input Linguistic: removes all focal elements, and removes all rules.

Figure 2-7. The Menu Item Help|Changes to an Analysis

The "**Help| Changes to an Analysis**" dialog in Figure 2-7 describes how changes to an analysis change the state of that analysis. LinguisticBelief recursively propagates any change to an analysis throughout the entire analysis, saving portions of the analysis not affected by the change. For example, deletion of a linguistic variable removes that variable and removes all rules for any rule-based linguistic that requires the deleted variable as an input. Changing the name of a linguistic variable does not change the rules that use that variable.

2.2.3 Utilities Menu

The items under the Utilities menu are shown in Figure 2-8.

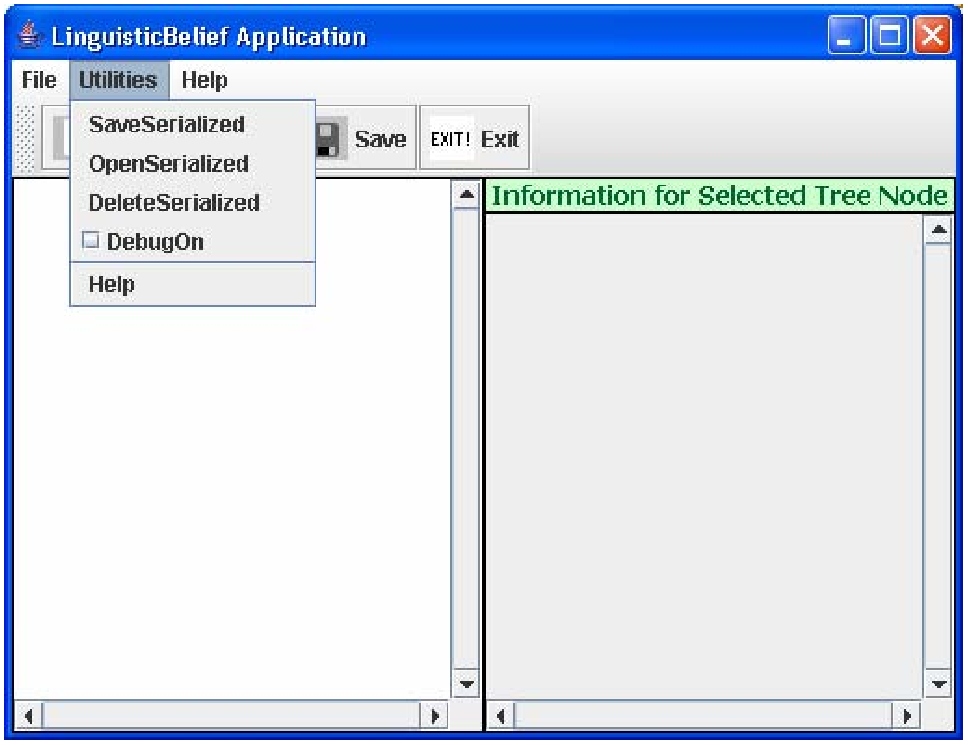
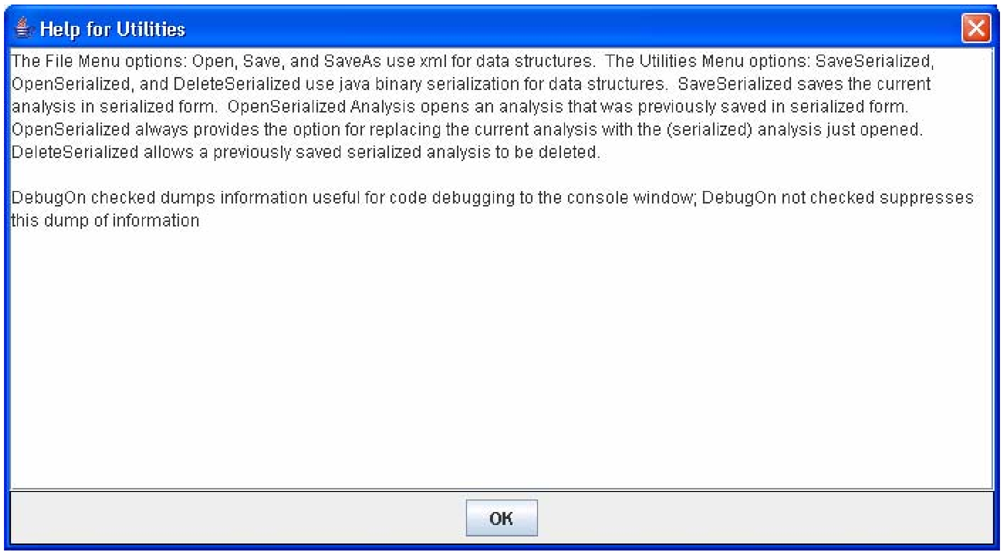


Figure 2-8. Options Under the Utilities Menu

The **Utilities|Help** choice displays information describing the other options under Utilities, as shown in Figure 2-9.



The File Menu options: Open, Save, and SaveAs use xml for data structures. The Utilities Menu options: SaveSerialized, OpenSerialized, and DeleteSerialized use java binary serialization for data structures. SaveSerialized saves the current analysis in serialized form. OpenSerialized Analysis opens an analysis that was previously saved in serialized form. OpenSerialized always provides the option for replacing the current analysis with the (serialized) analysis just opened. DeleteSerialized allows a previously saved serialized analysis to be deleted.

DebugOn checked dumps information useful for code debugging to the console window; DebugOn not checked suppresses this dump of information

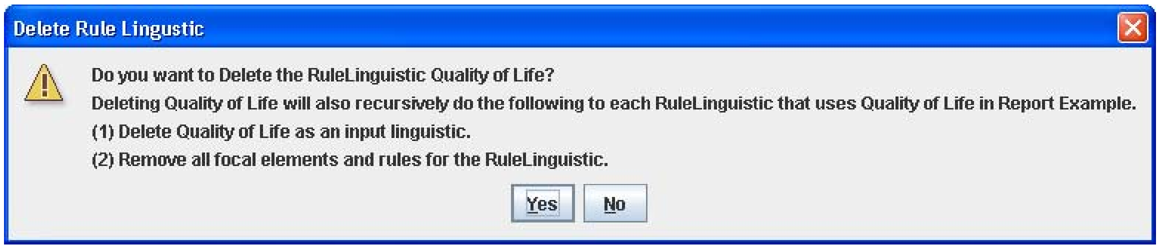
Figure 2-9. The Menu Item Utilities|Help

As indicated in Figure 2-9, Utilities options include:

- **Utilities|SaveSerialized** saves the current analysis in Java serialized (binary) form
- **Utilities|OpenSerialized** loads an existing analysis that was saved in serialized form.
- **Utilities|DeleteSerialized** deletes an existing analysis that was saved in serialized form.

As discussed in Appendix A, LinguisticBelief has the capability to save and load an analysis in serialized form as well as in xml format. The **Save, SaveAs**, and **Open** options previously discussed use the xml format.

- Check **Utilities|DebugOn** to provide information dialogs and dump detailed information to the console window (System.out). For example, with DebugOn selected, if the user tries to delete the linguistic variable "Quality of Life" for the example analysis in Section 1, the code provides the warning shown in Figure 2-10.


Delete Rule Lingustic

Do you want to Delete the RuleLinguistic Quality of Life?
Deleting Quality of Life will also recursively do the following to each RuleLinguistic that uses Quality of Life in Report Example.
(1) Delete Quality of Life as an input linguistic.
(2) Remove all focal elements and rules for the RuleLinguistic.

Figure 2-10. Example of Warning with Utilities|DebugOn Selected

If the user selects "Yes" in the dialog in Figure 2-10, the code will delete the rule-based linguistic "Quality of Life" and dump the information shown in Listing 2-1 to the console. (Note the effect of recursion as discussed in Appendix A.)

Listing 2-1. Example of Information Provided when Removing a Linguistic Variable with Utilities|DebugOn Selected

```
RuleLinguisticDialog is deleting Quality of Life from RuleLinguistic Quality of Life
   deleteThisAndAllRulesAndInputLinguisticIfInputLinguistic() looking at: Quality of Life

RuleLinguisticDialog is deleting Quality of Life from RuleLinguistic Happiness
   deleteThisAndAllRulesAndInputLinguisticIfInputLinguistic() looking at: Happiness
      deleteThisAndAllRulesAndInputLinguisticIfInputLinguistic() operating recursively on: Quality of Life
   deleteAllRulesAndInputLinguisticIfInputLinguistic() looking at: Quality of Life
         deleteThisAndAllRulesAndInputLinguisticIfInputLinguistic() removing Quality of Life
            deleteThisAndAllRulesAndInputLinguisticIfInputLinguistic() removed focal elements from
Happiness
            deleteThisAndAllRulesAndInputLinguisticIfInputLinguistic() removed rules from Happiness
```

2.3 Creating an Analysis

This section shows how to create an analysis by example, using the same one as discussed in Section 1.1. The procedure to create a New analysis follows:

- Click on the **New** button.
- Enter the name of the New analysis. If the user types the same name as the name of a previously saved analysis, the user has the option to replace the older analysis with the new one. Assuming that a new analysis named “Section 1 Example” is created, the gui appears as in Figure 2-11; an analysis named “Section 1 Example” has been created, with no linguistic variables as yet.

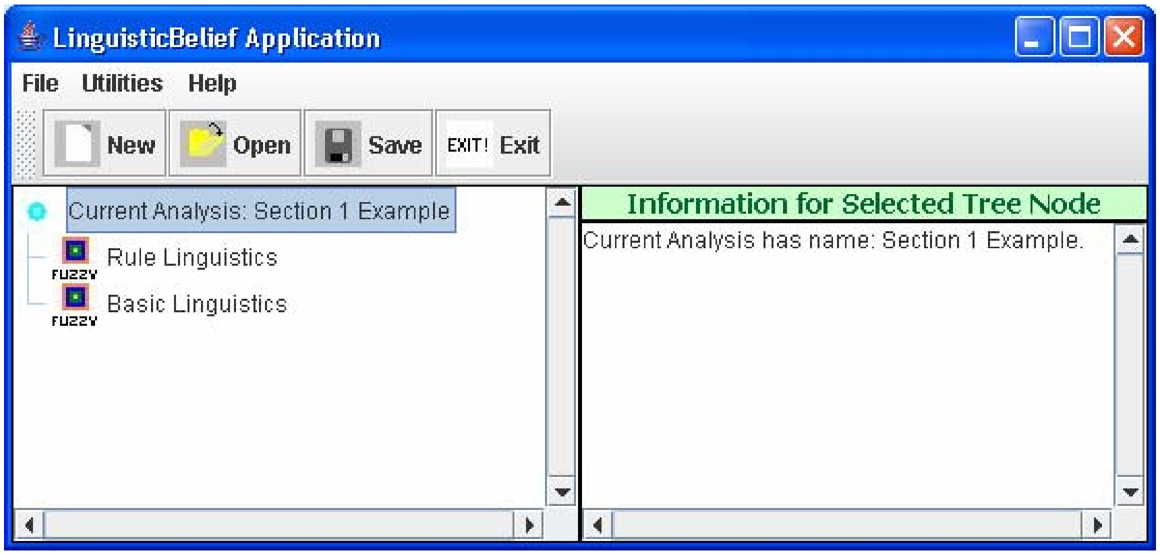
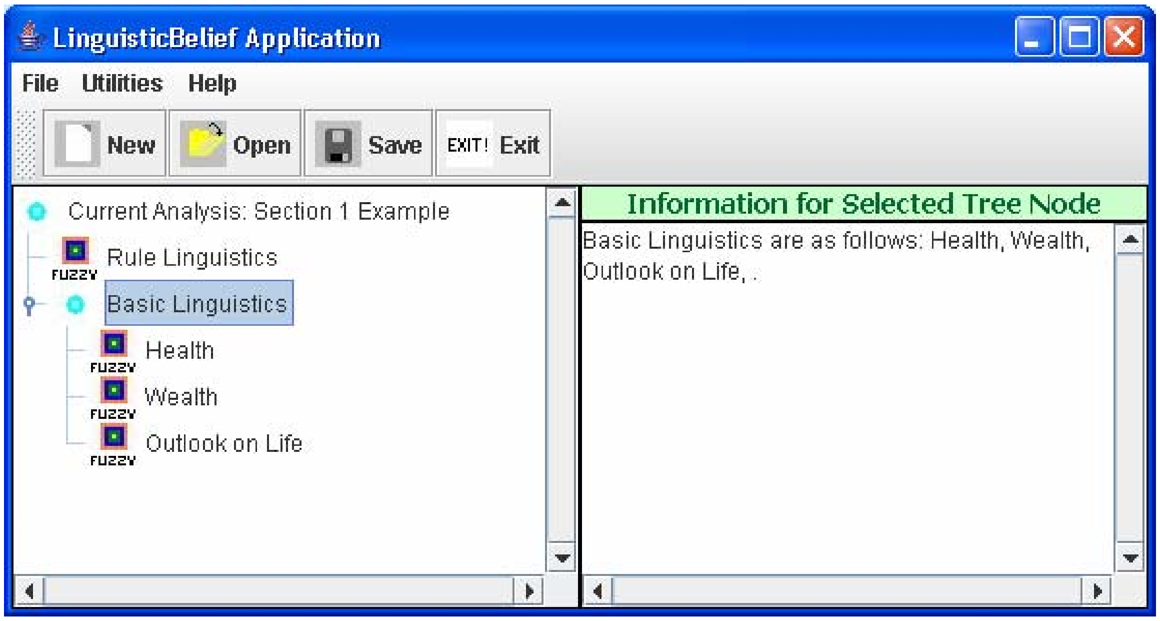

LinguisticBelief Application
File
Utilities
Help
New
Open
Save
EXIT! Exit
Current Analysis: Section 1 Example
Rule Linguistics
Basic Linguistics
Information for Selected Tree Node
Current Analysis has name: Section 1 Example.

Figure 2-11. “Section 1 Example” with No Linguistic Variables

A basic linguistic is a linguistic to which evidence will be assigned. A rule-based linguistic is a linguistic formed from the combination of two or more other linguistics (basic or rule-based). LinguisticBelief requires that the name for every linguistic, basic or rule-based, be unique.

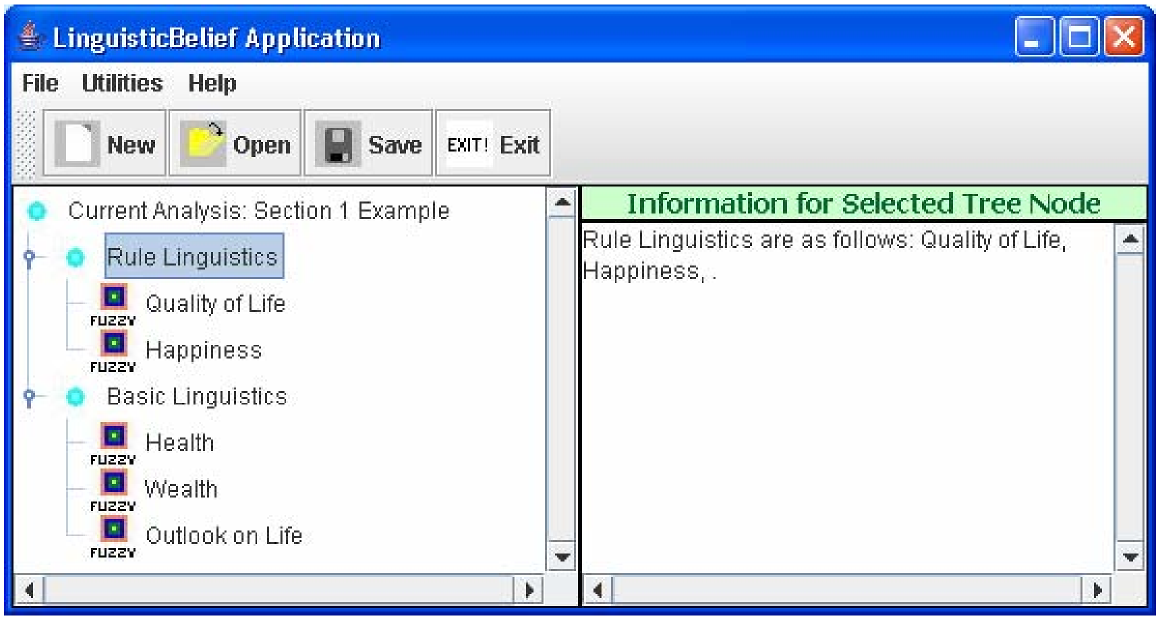
- To **add a basic linguistic**, left-click the Basic Linguistics node in the tree and enter the name of the basic linguistic to be added.
- To **add a rule-based linguistic**, left-click the Rule Linguistics node in the tree and entering the name of the rule-based linguistic to be added.

Figure 2-12 shows the results for the “Section 1 Example” after adding the Basic Linguistics “Health,” “Wealth,” and “Outlook on Life.” Figure 2-13 shows the results for the “Section 1 Example” after adding the Rule Linguistics “Quality of Life” and “Happiness.”



Basic Linguistics are as follows: Health, Wealth, Outlook on Life, .

Figure 2-12. "Section 1 Example" After Adding Basic Linguistics



Rule Linguistics are as follows: Quality of Life, Happiness, .

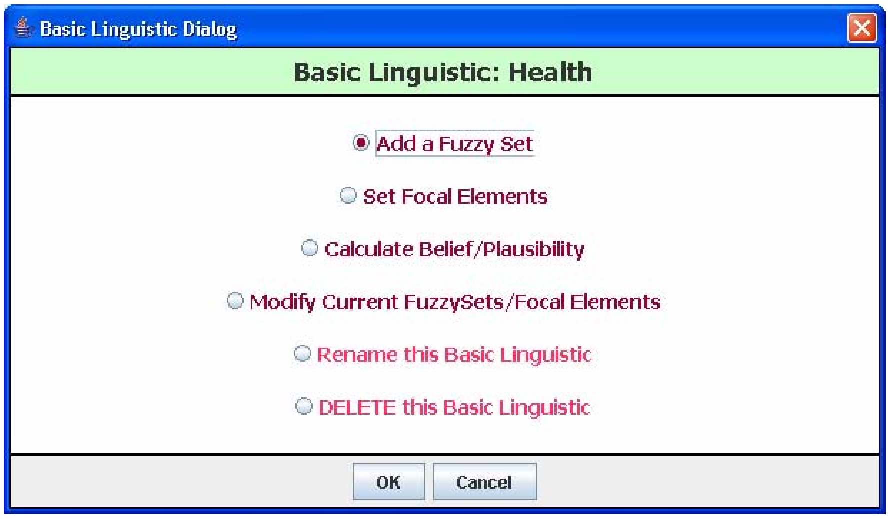
Figure 2-13. "Section 1 Example" After Adding Rule Linguistics

The next step is to operate on the Rule Linguistics and the Basic Linguistics you have created:

- Left click a linguistic variable (basic or rule) to highlight it.
- Right click on the selected variable to display the operations allowed for that variable. For example, Figure 2-14 shows the Basic Linguistic options available for "Health." Figure 2-17 shows the Rule Linguistic options available for "Quality of Life."

2.3.1 Adding Fuzzy Sets

Next, the user adds fuzzy sets for **all** variables, both basic and rule. Click on the **Add a Fuzzy Set** radio button as shown in Figure 2-14 for a basic linguistic and Figure 2-15 for a rule linguistic.

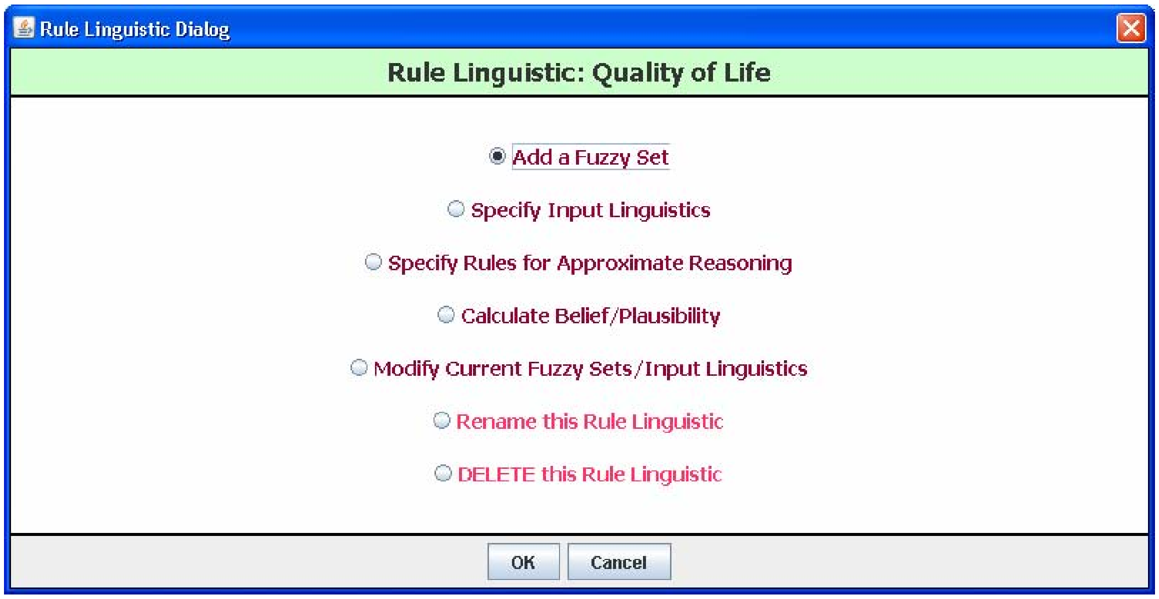


Basic Linguistic Dialog

Basic Linguistic: Health

[✓ Add a Fuzzy Set]
Set Focal Elements
Calculate Belief/Plausibility
Modify Current FuzzySets/Focal Elements
Rename this Basic Linguistic
DELETE this Basic Linguistic

Figure 2-14. Options for the Basic Linguistic "Health"



Rule Linguistic Dialog

Rule Linguistic: Quality of Life

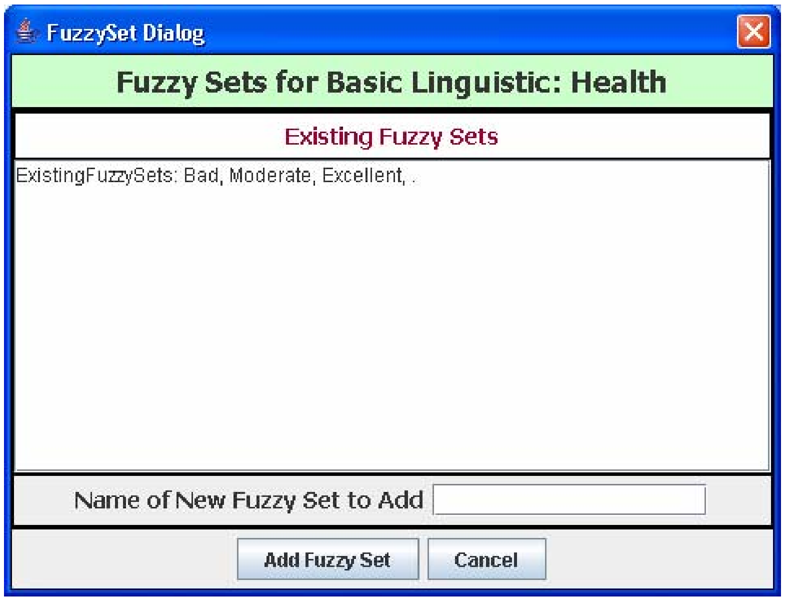
[✓ Add a Fuzzy Set]
Specify Input Linguistics
Specify Rules for Approximate Reasoning
Calculate Belief/Plausibility
Modify Current Fuzzy Sets/Input Linguistics
Rename this Rule Linguistic
DELETE this Rule Linguistic

Figure 2-15. Options for the Rule Linguistic "Quality of Life"

Figure 2-16 shows the dialog for adding fuzzy sets to "Health" after **all** the fuzzy sets have been added.

- Type each fuzzy set in the **Name of New Fuzzy Set to Add** edit box and clicks **Add Fuzzy Set** button. Type each one separately; do not type commas.

- Add fuzzy sets for all the other linguistic variables in a similar manner. The system displays the fuzzy sets as you add them.

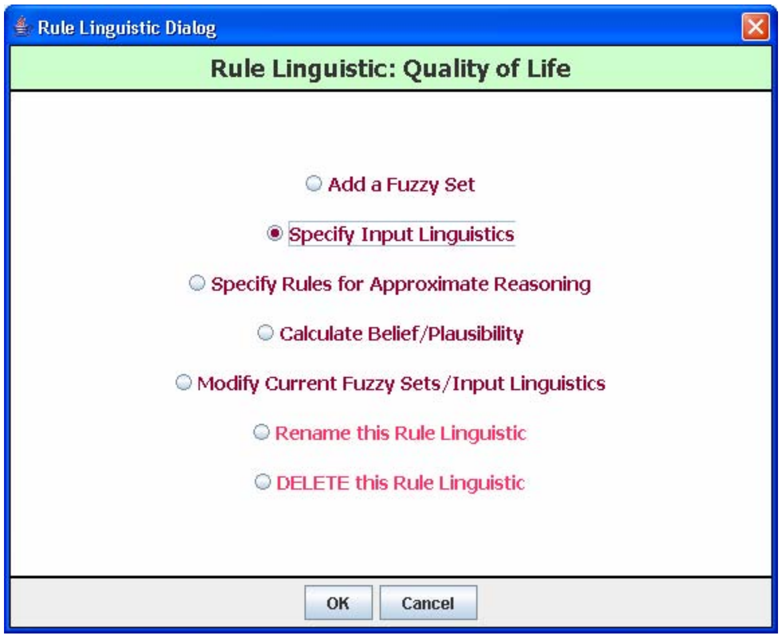

FuzzySet Dialog
Fuzzy Sets for Basic Linguistic: Health
Existing Fuzzy Sets
Name of New Fuzzy Set to Add
Add Fuzzy Set
Cancel

ExistingFuzzySets: Bad, Moderate, Excellent, .

Figure 2-16. Fuzzy Sets for "Health"

2.3.2 Specify Input Linguistics

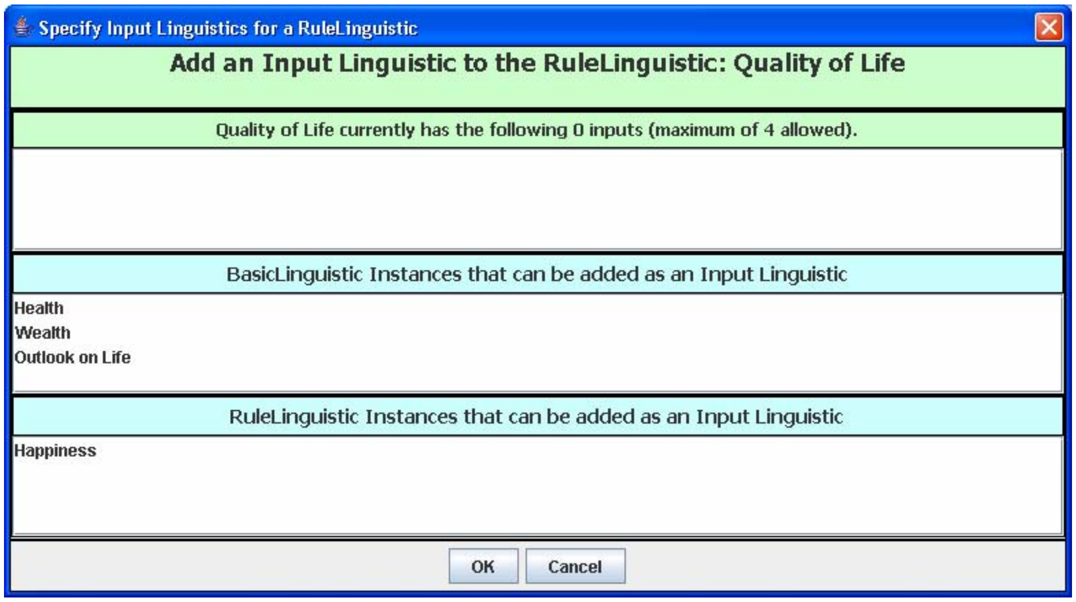
Once all the fuzzy sets for each linguistic variable (basic and rule-based) have been added, the user will specify the input linguistics for **each** rule-based linguistic. For example, selecting and right-clicking on "Quality of Life" shows the options in Figure 2-17.



- ○ Add a Fuzzy Set
- ● Specify Input Linguistics
- ○ Specify Rules for Approximate Reasoning
- ○ Calculate Belief/Plausibility
- ○ Modify Current Fuzzy Sets/Input Linguistics
- ○ Rename this Rule Linguistic
- ○ DELETE this Rule Linguistic

Figure 2-17. Specifying Input Linguistics for the Rule Linguistic "Quality of Life"

Selecting the **Specify Input Linguistics** button (as indicated in Figure 2-17), a dialog displays. The user clicks the rule and basic variables needed as inputs. As the user clicks a variable, the system moves the variables to the top panel automatically. Figure 2-18 shows this dialog for "Quality of Life" before the input linguistics have been specified and Figure 2-19 shows this dialog after the appropriate input linguistics have been clicked.


Specify Input Linguistics for a RuleLinguistic

Add an Input Linguistic to the RuleLinguistic: Quality of Life

Quality of Life currently has the following 0 inputs (maximum of 4 allowed).

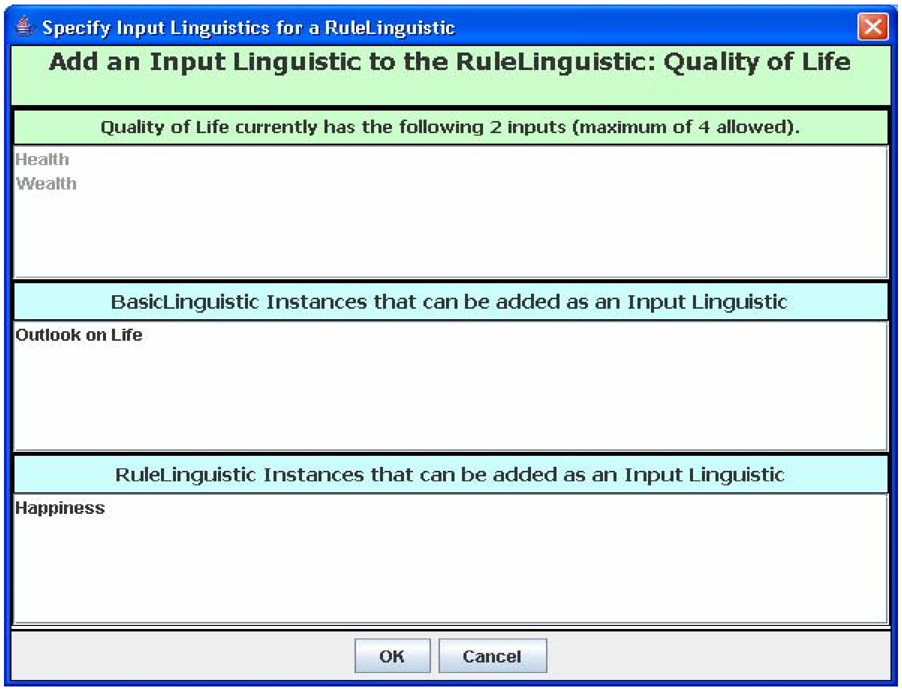
BasicLinguistic Instances that can be added as an Input Linguistic

Health
Wealth
Outlook on Life

RuleLinguistic Instances that can be added as an Input Linguistic

Happiness

Figure 2-18. Dialog for Adding Input Linguistics to "Quality of Life"


Specify Input Linguistics for a RuleLinguistic

Add an Input Linguistic to the RuleLinguistic: Quality of Life

Quality of Life currently has the following 2 inputs (maximum of 4 allowed).

Health
Wealth

BasicLinguistic Instances that can be added as an Input Linguistic

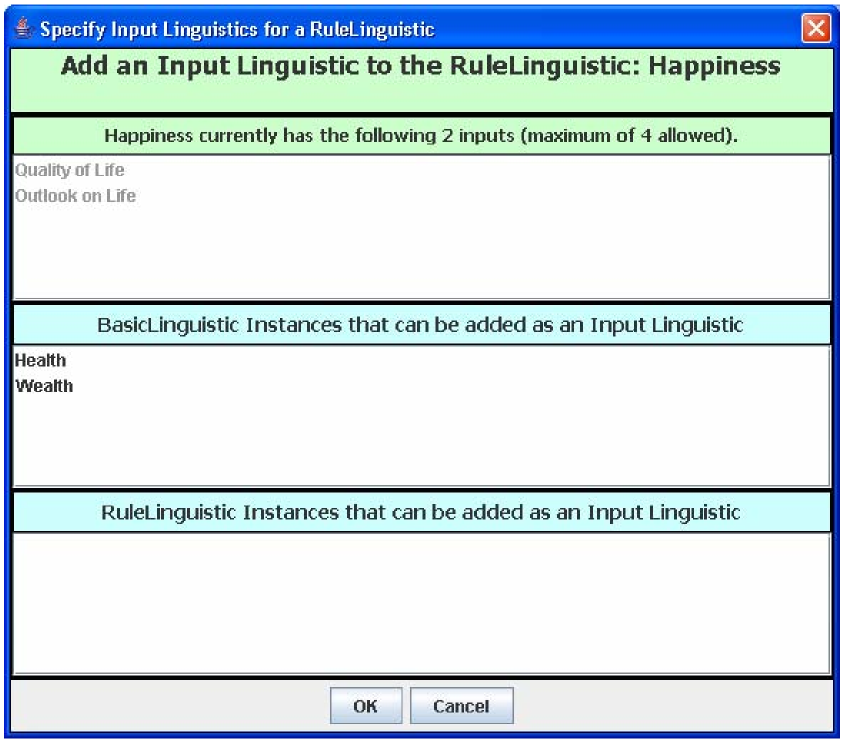
Outlook on Life

RuleLinguistic Instances that can be added as an Input Linguistic

Happiness

Figure 2-19. "Quality of Life" with Input Linguistics Specified

Similarly, the user selects input linguistics for "Happiness" as shown in Figure 2-20.



Specify Input Linguistics for a RuleLinguistic

Add an Input Linguistic to the RuleLinguistic: Happiness

Happiness currently has the following 2 inputs (maximum of 4 allowed).

Quality of Life
Outlook on Life

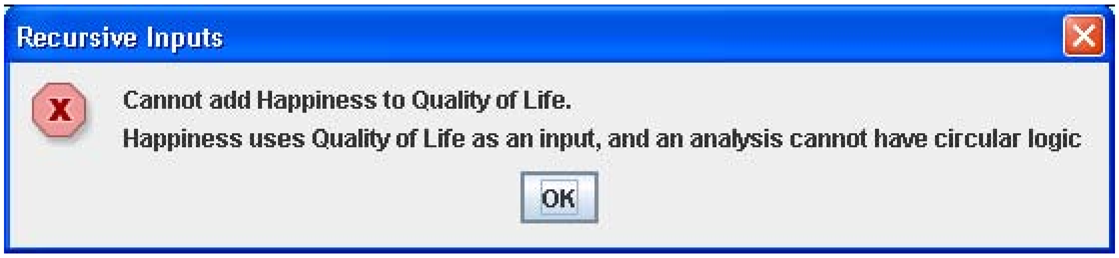
BasicLinguistic Instances that can be added as an Input Linguistic

Health
Wealth

RuleLinguistic Instances that can be added as an Input Linguistic

Figure 2-20. "Happiness" with Input Linguistics Specified

If we try to add "Happiness" as another input linguistic for "Quality of Life" we get the error message shown in Figure 2-21, because "Happiness" itself has 'Quality of Life" as an input and we cannot have circular logic.



Recursive Inputs

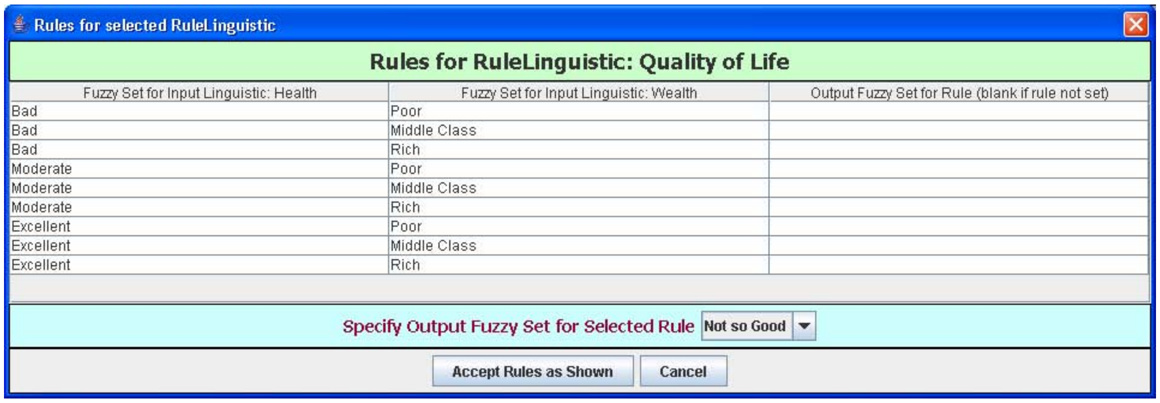
Cannot add Happiness to Quality of Life.
Happiness uses Quality of Life as an input, and an analysis cannot have circular logic

Figure 2-21. Error Message if Circular Logic Is Detected

2.4 Creating the Rule Base

Once all the basic and rule-based linguistics have been added, fuzzy sets added, and inputs for rule-based linguistics specified, the user will create rules for **all** the rule-based linguistics. Figure 2-22 shows the dialog for adding rules to "Quality of Life." Note that all the "Output Fuzzy Set for Rule (blank if rule not set)" entries are blank, indicating that no rules have been specified. Figure 2-23 shows this dialog after all the rules have been added. To add an Output Fuzzy Set for Rule:

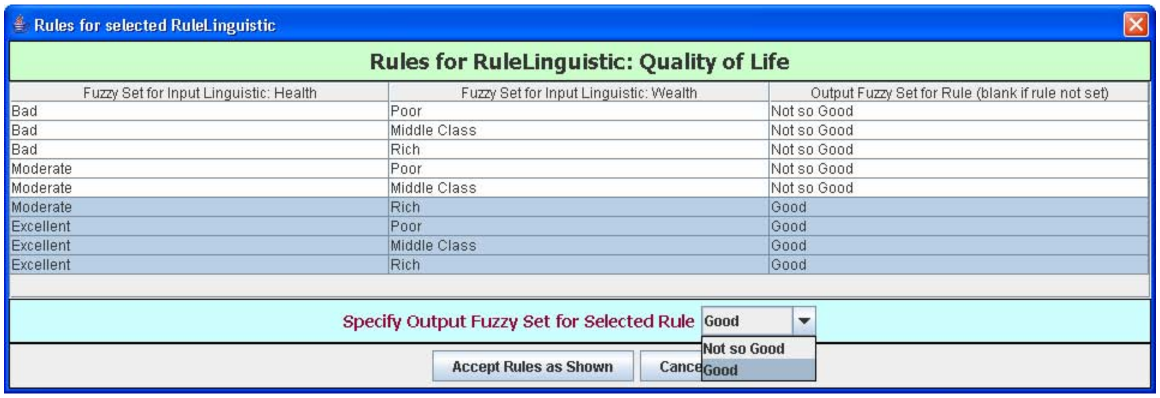
- Click on a blank line in the **Output Fuzzy Set for Rule** panel. (You may use standard Windows rules to select multiple lines to populate with the same selection.)
- Click on the combo box (**Specify Output Fuzzy Set for Selected Rule**) to display the list of fuzzy sets and select one. The blank line will be populated with your selection.
- When you are done, click on **Accept Rules as Shown**. You may accept a partial rule set, but all rules must be completed for a rule linguistic before you can calculate the result for that linguistic (Refer to Section 2.6.)
- If you click **Cancel**, you can revert to the previous rule set state without saving any changes since the last time you clicked **Accept Rules as Shown**.


Rules for selected RuleLinguistic
Specify Output Fuzzy Set for Selected Rule
Not so Good
Accept Rules as Shown
Cancel

Rules for RuleLinguistic: Quality of Life

Fuzzy Set for Input Linguistic: Health	Fuzzy Set for Input Linguistic: Wealth	Output Fuzzy Set for Rule (blank if rule not set)
Bad	Poor	
Bad	Middle Class	
Bad	Rich	
Moderate	Poor	
Moderate	Middle Class	
Moderate	Rich	
Excellent	Poor	
Excellent	Middle Class	
Excellent	Rich	

Figure 2-22. Empty Rule Base for 'Quality of Life"

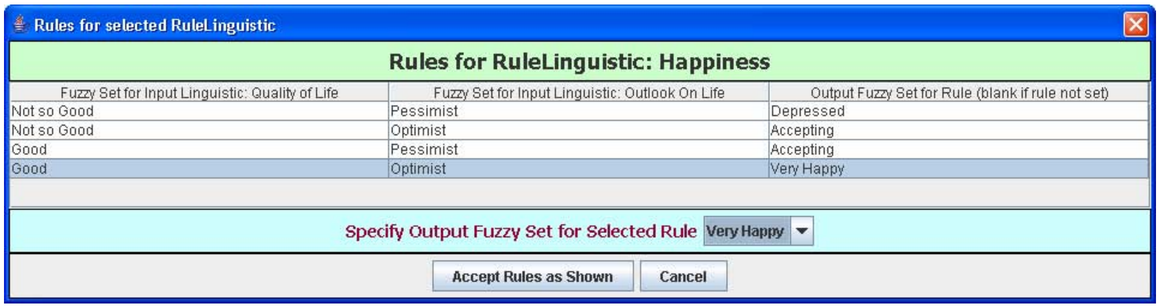

Rules for selected RuleLinguistic
Specify Output Fuzzy Set for Selected Rule
Good
Not so Good
Good
Accept Rules as Shown
Cancel

Rules for RuleLinguistic: Quality of Life

Fuzzy Set for Input Linguistic: Health	Fuzzy Set for Input Linguistic: Wealth	Output Fuzzy Set for Rule (blank if rule not set)
Bad	Poor	Not so Good
Bad	Middle Class	Not so Good
Bad	Rich	Not so Good
Moderate	Poor	Not so Good
Moderate	Middle Class	Not so Good
Moderate	Rich	Good
Excellent	Poor	Good
Excellent	Middle Class	Good
Excellent	Rich	Good

Figure 2-23. Complete Rule Base for "Quality of Life"

Similarly, the user adds the rules for "Happiness" as shown in Figure 2-24.



Rules for RuleLinguistic: Happiness

Fuzzy Set for Input Linguistic: Quality of Life	Fuzzy Set for Input Linguistic: Outlook On Life	Output Fuzzy Set for Rule (blank if rule not set)
Not so Good	Pessimist	Depressed
Not so Good	Optimist	Accepting
Good	Pessimist	Accepting
Good	Optimist	Very Happy

Specify Output Fuzzy Set for Selected Rule: Very Happy

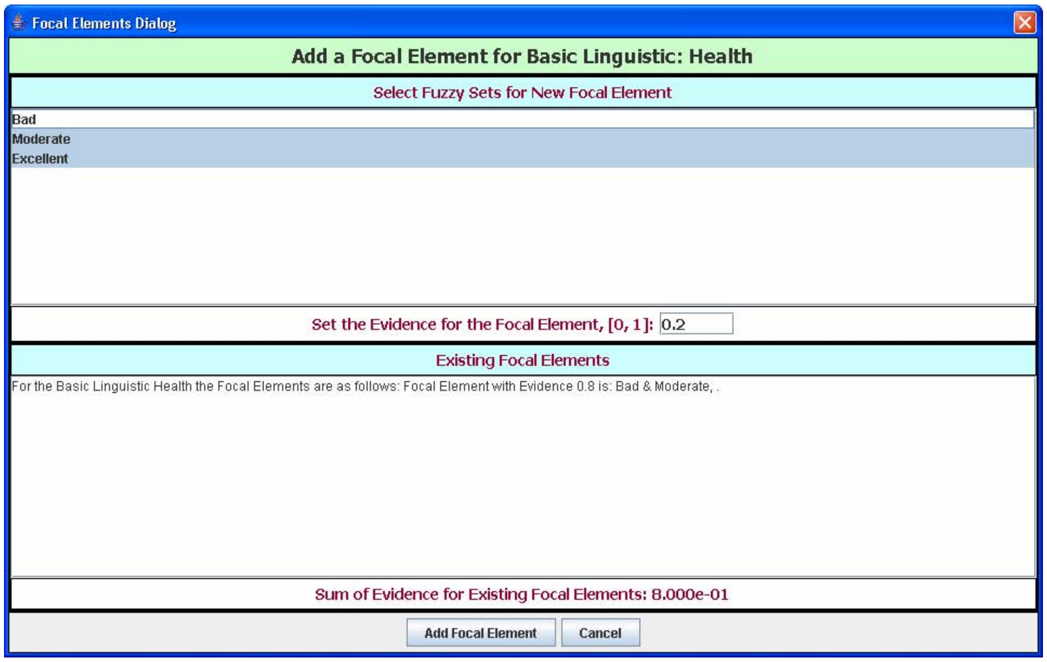
Figure 2-24. Complete Rule Base for "Happiness"

2.5 Add Focal Elements

At this stage, the "Section 1 Example" model is complete and ready for use. To perform the evaluation for a specific situation, the evidence for that situation–the focal elements–must be specified. For our example analysis, Section 1.1 provided the focal elements for evaluating a specific individual, "John." **Note**: The specific person's name is not entered by the user in the application. To specifically identify any person (i.e., "John"), you can use the **Save As** option to copy the model under the name "John." Now you change the evidence on the copied model "John" as discussed in this section. This allows you to separate your model from the specific situations to be evaluated.

To add focal elements:

- Right-click on a specific basic linguistic (Figure 2-13), e.g., "Health."
- Select the **Set Focal Elements** button (Figure 2-14) to display a dialog that allows the user to add focal elements.
- Select one or fuzzy sets from the top panel and enter the evidence for the selected sets in the **Set the Evidence for the Focal Element** edit box. Repeat this step for each focal element (the combined evidence for all focal elements must sum to "1"). Figure 2-25 shows the dialog for adding the focal elements for "Health" where one focal element has been added: {"Bad," "Moderate"} with evidence 0.8 and another focal element: {"Moderate," "Excellent"} with evidence 0.2 is in the process of being added.
- After all focal elements are added, click **Cancel** to exit.

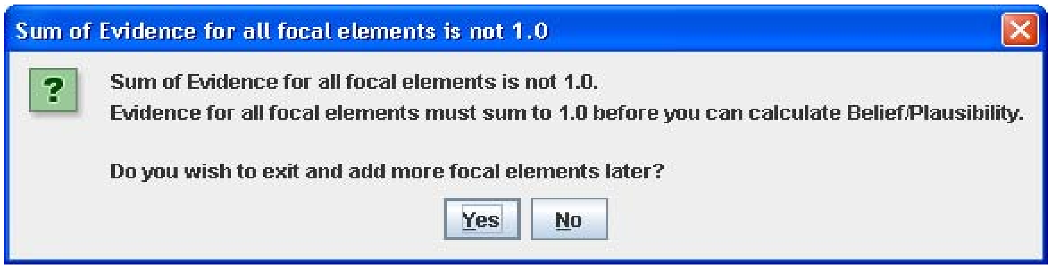

Focal Elements Dialog
Add a Focal Element for Basic Linguistic: Health
Select Fuzzy Sets for New Focal Element
Bad
Moderate
Excellent
Set the Evidence for the Focal Element, [0, 1]: 0.2
Existing Focal Elements
Add Focal Element
Cancel

For the Basic Linguistic Health the Focal Elements are as follows: Focal Element with Evidence 0.8 is: Bad & Moderate, .

Sum of Evidence for Existing Focal Elements: 8.000e-01

Figure 2-25. Adding Focal Elements for "Health" for "John"

If the user clicks on **Cancel** before adding elements whose evidence sums to 1.0, the warning dialog in Figure 2-26 displays.

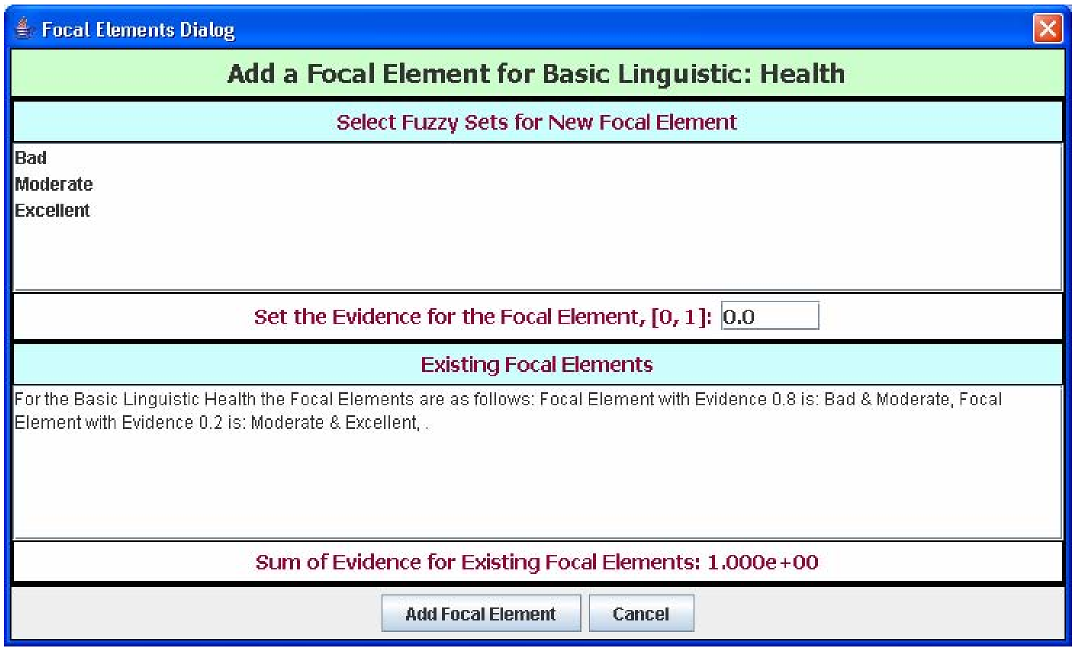


Sum of Evidence for all focal elements is not 1.0.
Evidence for all focal elements must sum to 1.0 before you can calculate Belief/Plausibility.

Do you wish to exit and add more focal elements later?

Figure 2-26. Warning of Incomplete Evidence for a Basic Linguistic

Figure 2-27 shows the result of adding all the focal elements to "Health."


Focal Elements Dialog
Add a Focal Element for Basic Linguistic: Health
Select Fuzzy Sets for New Focal Element
Bad
Moderate
Excellent
Set the Evidence for the Focal Element, [0, 1]:
Existing Focal Elements
Add Focal Element
Cancel

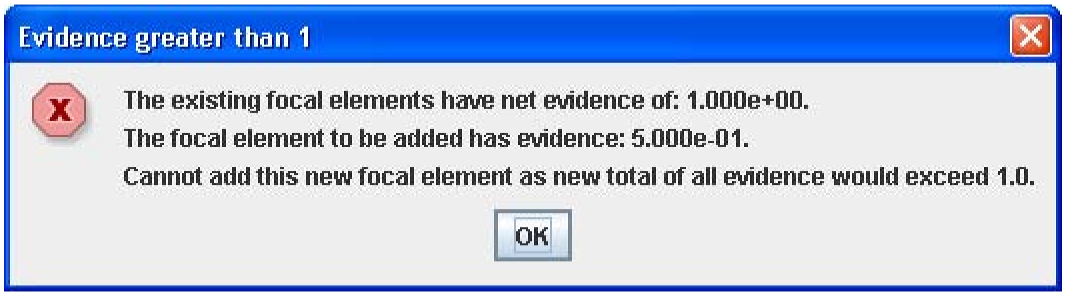
0.0

For the Basic Linguistic Health the Focal Elements are as follows: Focal Element with Evidence 0.8 is: Bad & Moderate, Focal Element with Evidence 0.2 is: Moderate & Excellent, .

Sum of Evidence for Existing Focal Elements: 1.000e+00

Figure 2-27. All Focal Elements Added for "Health" for "John"

Should the user attempt to add another focal element to "Health," the error dialog in Figure 2-28 is displayed as "Health" already has a set of focal elements whose evidence sums to 1.0.



The existing focal elements have net evidence of: 1.000e+00.
The focal element to be added has evidence: 5.000e-01.
Cannot add this new focal element as new total of all evidence would exceed 1.0.

Figure 2-28. Error Message for Attempt to Add a Focal Element that Would Result in Sum of Evidence over All Focal Elements Greater than One

To complete the body of evidence for "John," the focal elements for "Wealth" and "Outlook on Life" given in Section 1.1 are also added using the same process just discussed for adding the focal elements to "Health."

2.6 Calculate Results

Now, the results for any basic or rule linguistic for the specific situation can be calculated.
- Select and click any linguistic (basic or rule).
- Select **Calculate Belief/Plausibility** button (Figure 2-29).

For example, to calculate the results for “Quality of Life” for “John:”

- Select and right-click the “Quality of Life” node (Figure 2-13), resulting in the dialog shown in Figure 2-29.
- Select **Calculate Belief/Plausibility** button to display the results shown in Figure 2-30.

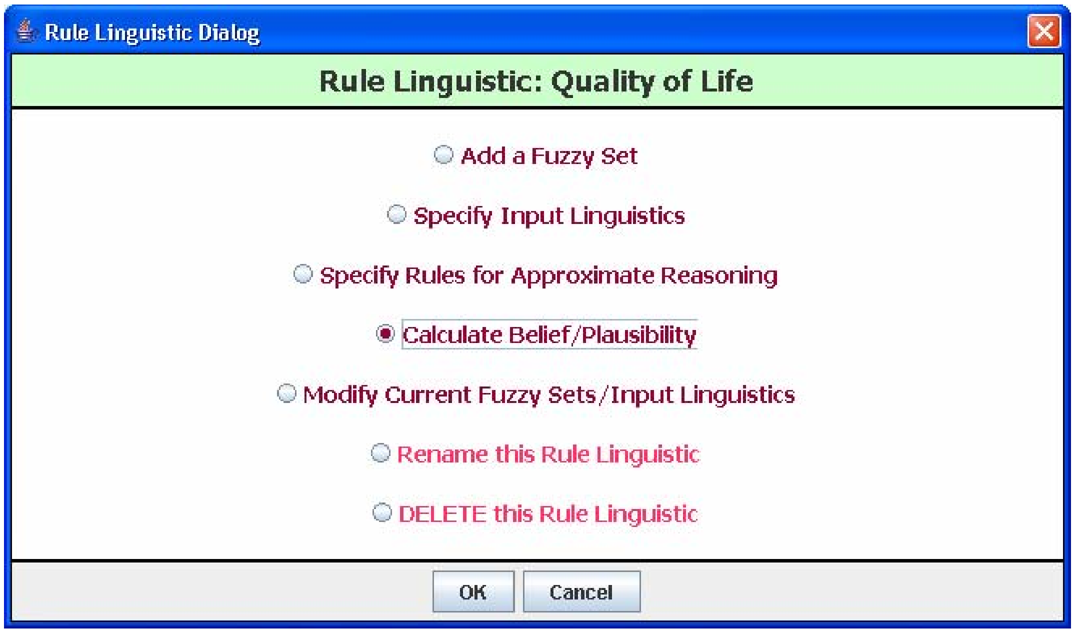
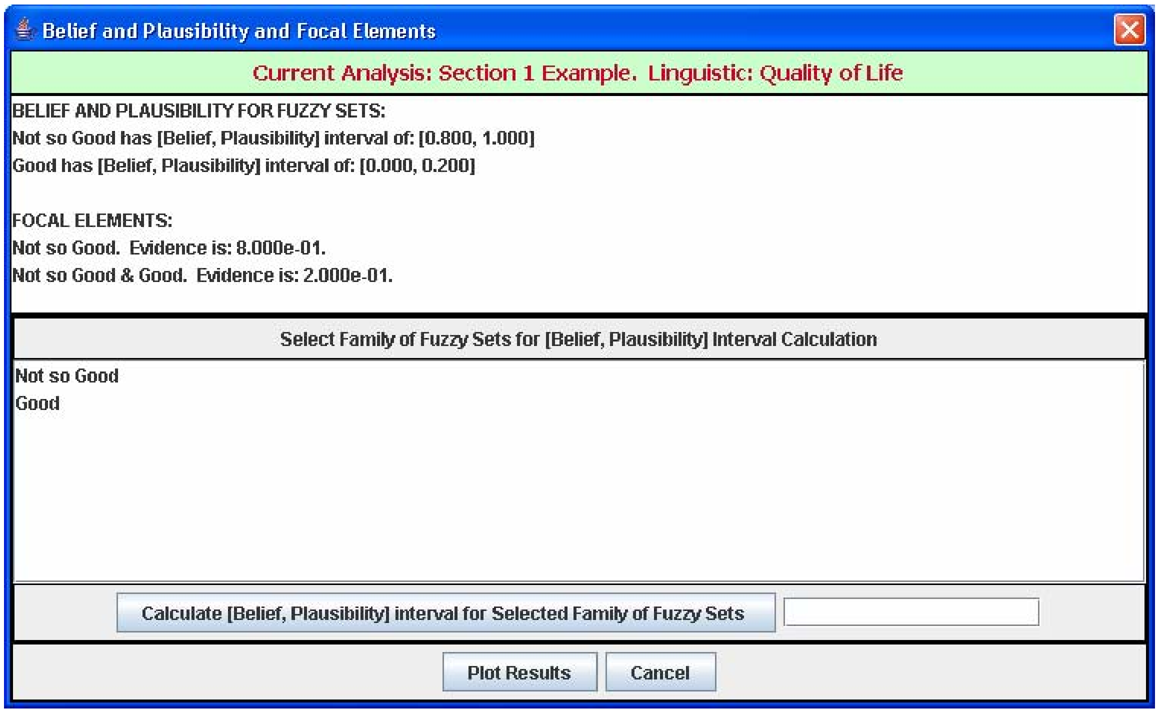

Rule Linguistic Dialog
Rule Linguistic: Quality of Life
Add a Fuzzy Set
Specify Input Linguistics
Specify Rules for Approximate Reasoning
[✓ Calculate Belief/Plausibility]
Modify Current Fuzzy Sets/Input Linguistics
Rename this Rule Linguistic
DELETE this Rule Linguistic
OK
Cancel

Figure 2-29. Generating Results for “Quality of Life” for “John”

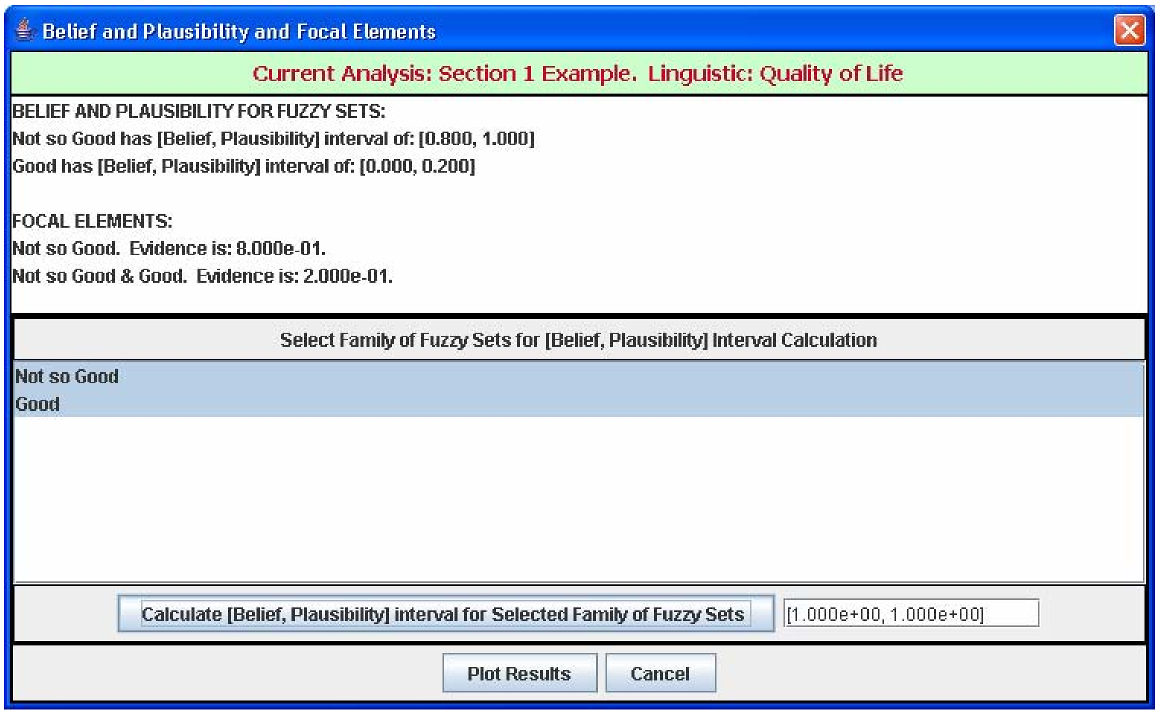

Belief and Plausibility and Focal Elements
Current Analysis: Section 1 Example. Linguistic: Quality of Life

```
BELIEF AND PLAUSIBILITY FOR FUZZY SETS:
Not so Good has [Belief, Plausibility] interval of: [0.800, 1.000]
Good has [Belief, Plausibility] interval of: [0.000, 0.200]

FOCAL ELEMENTS:
Not so Good.  Evidence is: 8.000e-01.
Not so Good & Good.  Evidence is: 2.000e-01.
```

Select Family of Fuzzy Sets for [Belief, Plausibility] Interval Calculation
Not so Good
Good
Calculate [Belief, Plausibility] interval for Selected Family of Fuzzy Sets
Plot Results
Cancel

Figure 2-30. Results for “Quality of Life” for “John”

In the results dialog of Figure 2-30, there is a button "Calculate [Belief, Plausibility] Interval for Selected Family of Fuzzy Sets." Figure 2-31 shows an example use of this capability for the selected family {"Not so Good," "Good"}; since this family is the entire universe of discourse for "Quality of Life," both belief and plausibility are 1.0, as expected.


Belief and Plausibility and Focal Elements
Current Analysis: Section 1 Example. Linguistic: Quality of Life

```
BELIEF AND PLAUSIBILITY FOR FUZZY SETS:
Not so Good has [Belief, Plausibility] interval of: [0.800, 1.000]
Good has [Belief, Plausibility] interval of: [0.000, 0.200]

FOCAL ELEMENTS:
Not so Good.  Evidence is: 8.000e-01.
Not so Good & Good.  Evidence is: 2.000e-01.
```

Select Family of Fuzzy Sets for [Belief, Plausibility] Interval Calculation
Not so Good
Good
Calculate [Belief, Plausibility] interval for Selected Family of Fuzzy Sets
[1.000e+00, 1.000e+00]
Plot Results
Cancel

Figure 2-31. "Quality of Life" for "John" with a Family Selected

- Click on the **Plot Results** button shown in the dialog in Figure 2-31, to display the dialog in Figure 2-32, requiring the user to order the fuzzy sets from "Best" to "Worst."

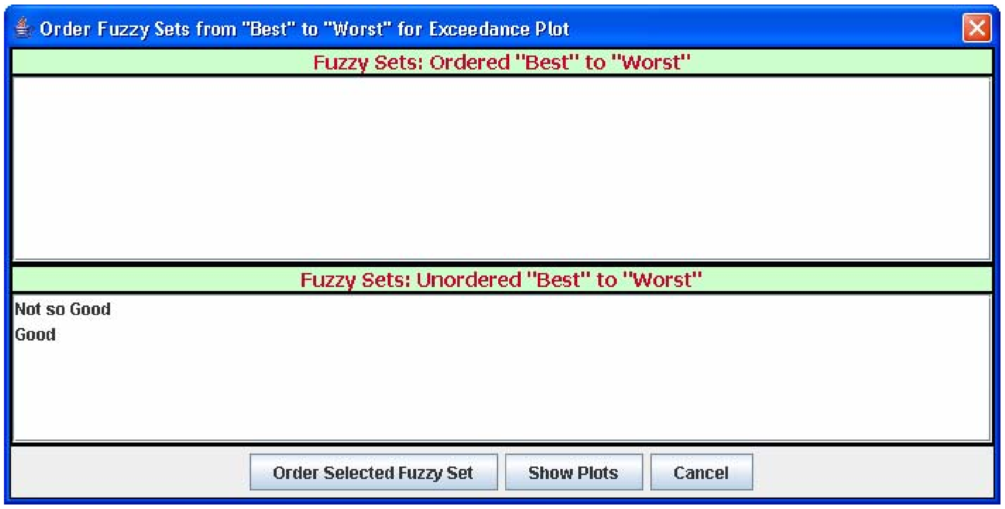

Order Fuzzy Sets from "Best" to "Worst" for Exceedance Plot
Fuzzy Sets: Ordered "Best" to "Worst"
Fuzzy Sets: Unordered "Best" to "Worst"
Not so Good
Good
Order Selected Fuzzy Set
Show Plots
Cancel

Figure 2-32. Fuzzy Sets, Not Ordered

- Click on the "Best" unordered fuzzy set in the lower panel. The fuzzy sets in the lower panel titled "Fuzzy Sets: Unordered "Best" to "Worst"" must be selected one at a time from "Best" to "Worst" and added to the upper panel titled 'Fuzzy Sets: Ordered "Best" to "Worst"'.
- Click on the **Order Selected Fuzzy Set** button. The selected will move from the "Unordered" panel to the "Ordered" panel.

Figure 2-33 shows the results of adding "Good," the "Best" fuzzy set, to the panel of ordered fuzzy sets.

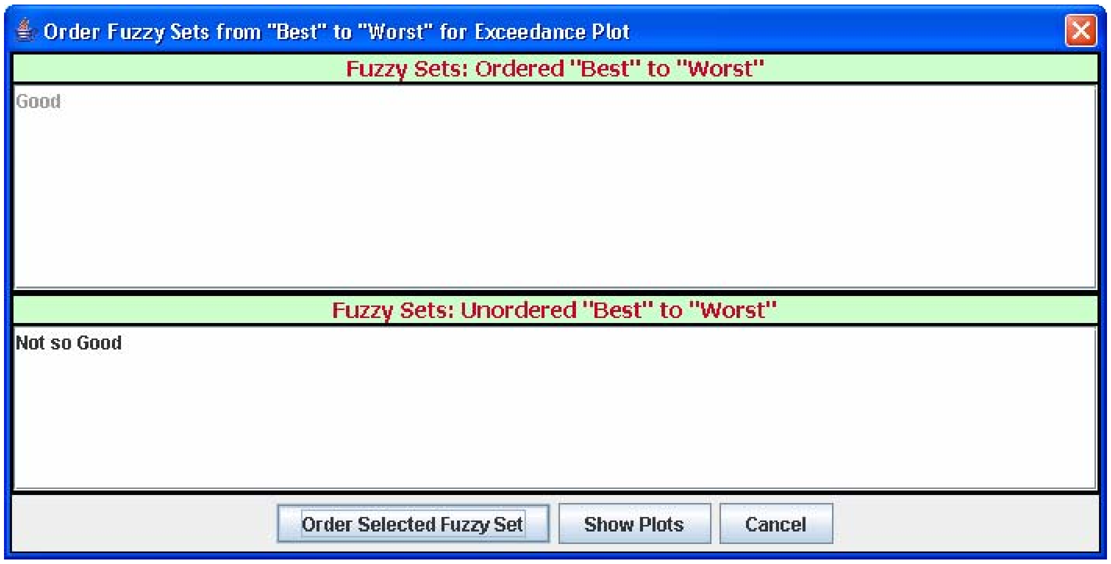

Order Fuzzy Sets from "Best" to "Worst" for Exceedance Plot
Fuzzy Sets: Ordered "Best" to "Worst"
Good
Fuzzy Sets: Unordered "Best" to "Worst"
Not so Good
Order Selected Fuzzy Set
Show Plots
Cancel

Figure 2-33. Fuzzy Sets, Partially Ordered

Figure 2-34 shows the results after all the fuzzy sets have been ordered from "Best" to "Worst."

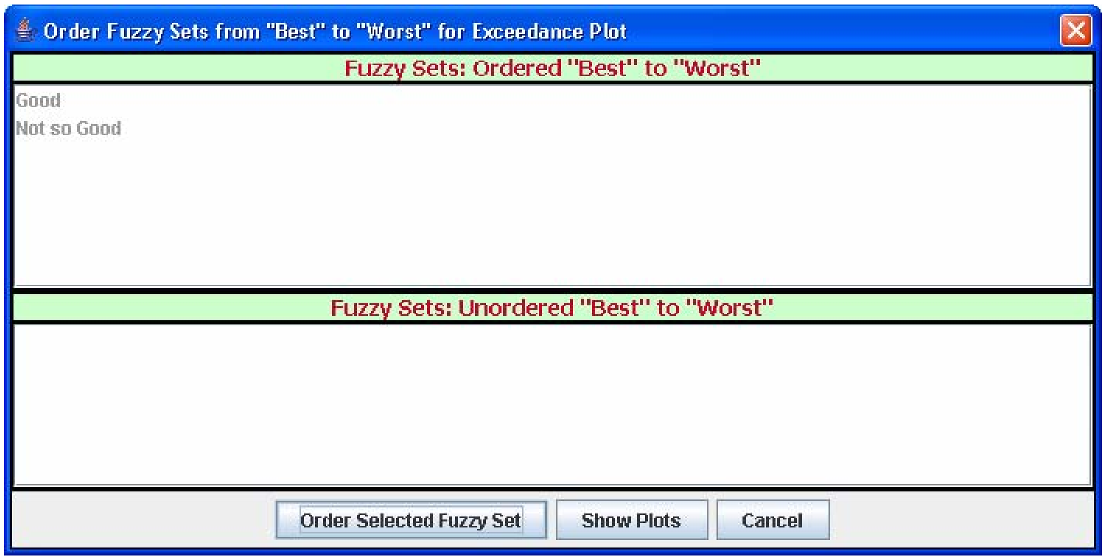

Order Fuzzy Sets from "Best" to "Worst" for Exceedance Plot
Fuzzy Sets: Ordered "Best" to "Worst"
Good
Not so Good
Fuzzy Sets: Unordered "Best" to "Worst"
Order Selected Fuzzy Set
Show Plots
Cancel

Figure 2-34. Fuzzy Sets, All Ordered

Once all the fuzzy sets have been ordered, click the **Show Plots** button in Figure 2-34 to display the graphical results of Figure 2-35.

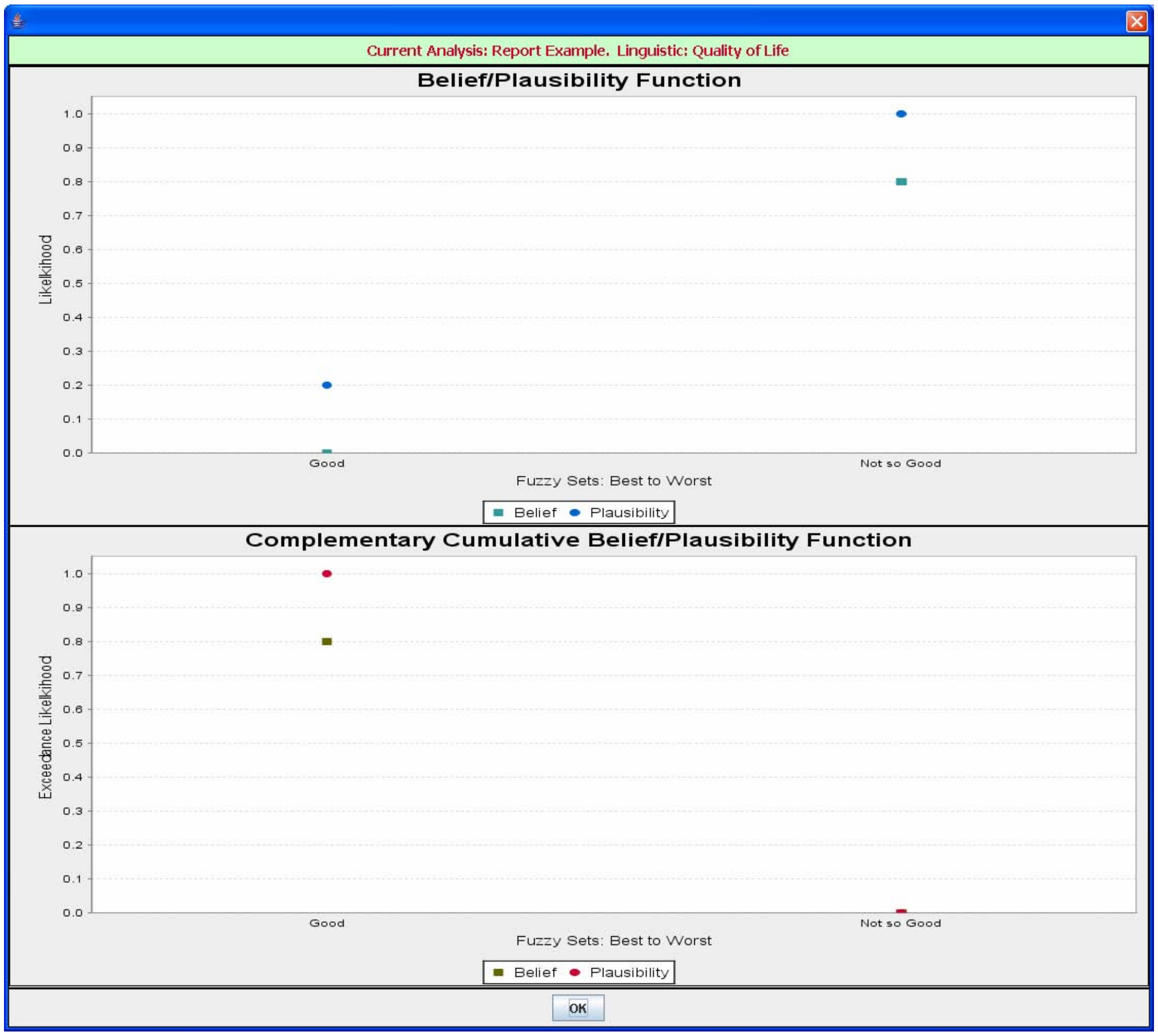
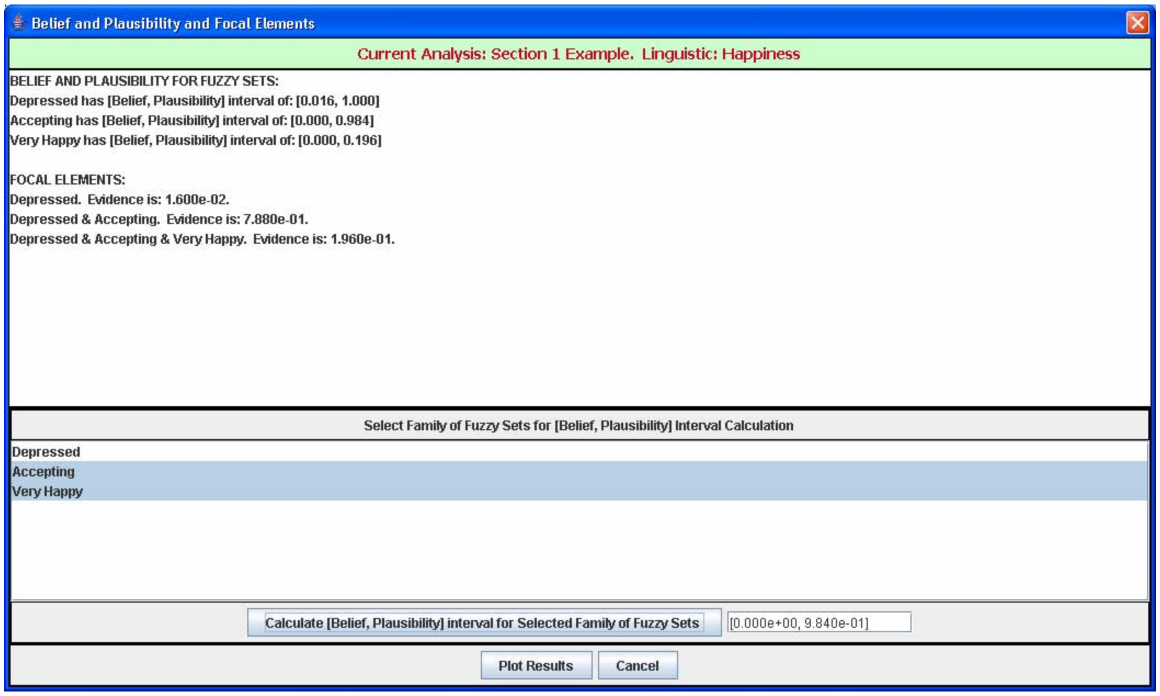

Current Analysis: Report Example. Linguistic: Quality of Life
Belief/Plausibility Function
Likelihood
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
Good
Not so Good
Fuzzy Sets: Best to Worst
■ Belief ● Plausibility
Complementary Cumulative Belief/Plausibility Function
Exceedance Likelihood
Good
Not so Good
Fuzzy Sets: Best to Worst
■ Belief ● Plausibility
OK

Figure 2-35. Graphical Results for "Quality of Life" for "John"

Similarly, the results for any of the other Rule Linguistics in the analysis can be generated. Figures 2-36 through 2-38 show results for "Happiness."


Belief and Plausibility and Focal Elements
Current Analysis: Section 1 Example. Linguistic: Happiness

```
BELIEF AND PLAUSIBILITY FOR FUZZY SETS:
Depressed has [Belief, Plausibility] interval of: [0.016, 1.000]
Accepting has [Belief, Plausibility] interval of: [0.000, 0.984]
Very Happy has [Belief, Plausibility] interval of: [0.000, 0.196]

FOCAL ELEMENTS:
Depressed.  Evidence is: 1.600e-02.
Depressed & Accepting.  Evidence is: 7.880e-01.
Depressed & Accepting & Very Happy.  Evidence is: 1.960e-01.
```

Select Family of Fuzzy Sets for [Belief, Plausibility] Interval Calculation
Depressed
Accepting
Very Happy
Calculate [Belief, Plausibility] interval for Selected Family of Fuzzy Sets
[0.000e+00, 9.840e-01]
Plot Results
Cancel

Figure 2-36. "Happiness" for "John"

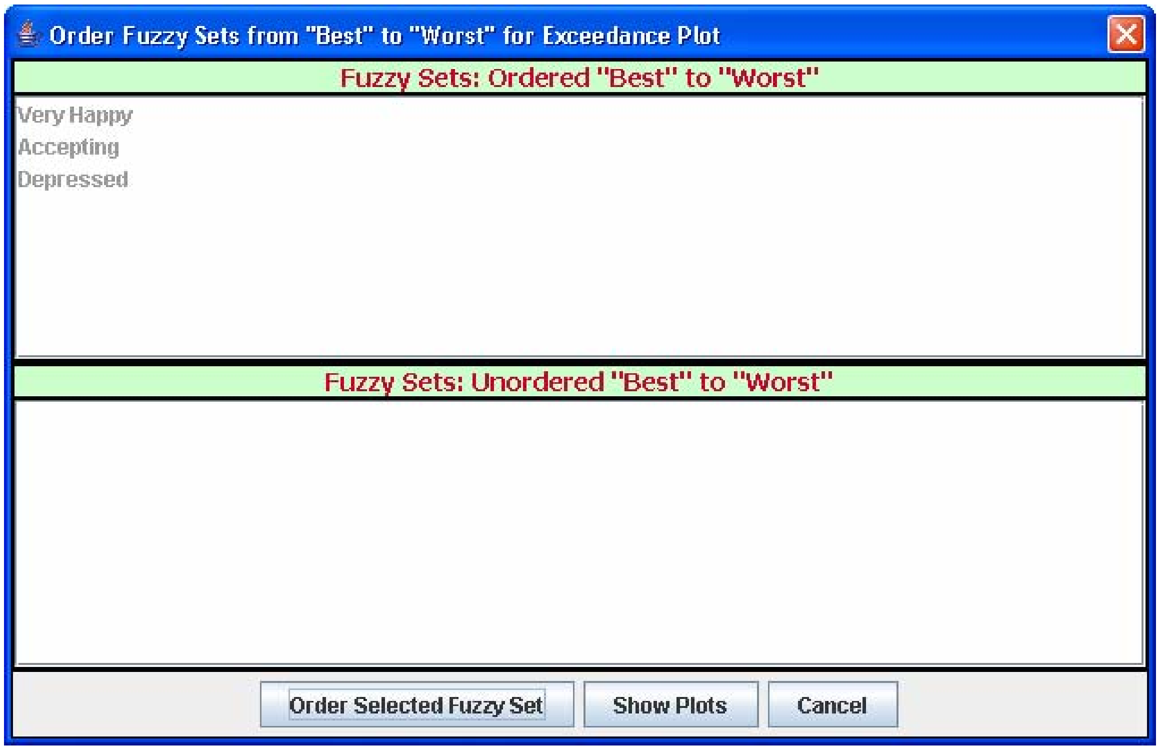

Order Fuzzy Sets from "Best" to "Worst" for Exceedance Plot
Fuzzy Sets: Ordered "Best" to "Worst"
Very Happy
Accepting
Depressed
Fuzzy Sets: Unordered "Best" to "Worst"
Order Selected Fuzzy Set
Show Plots
Cancel

Figure 2-37. "Happiness" Fuzzy Sets Ordered "Best" to "Worst"

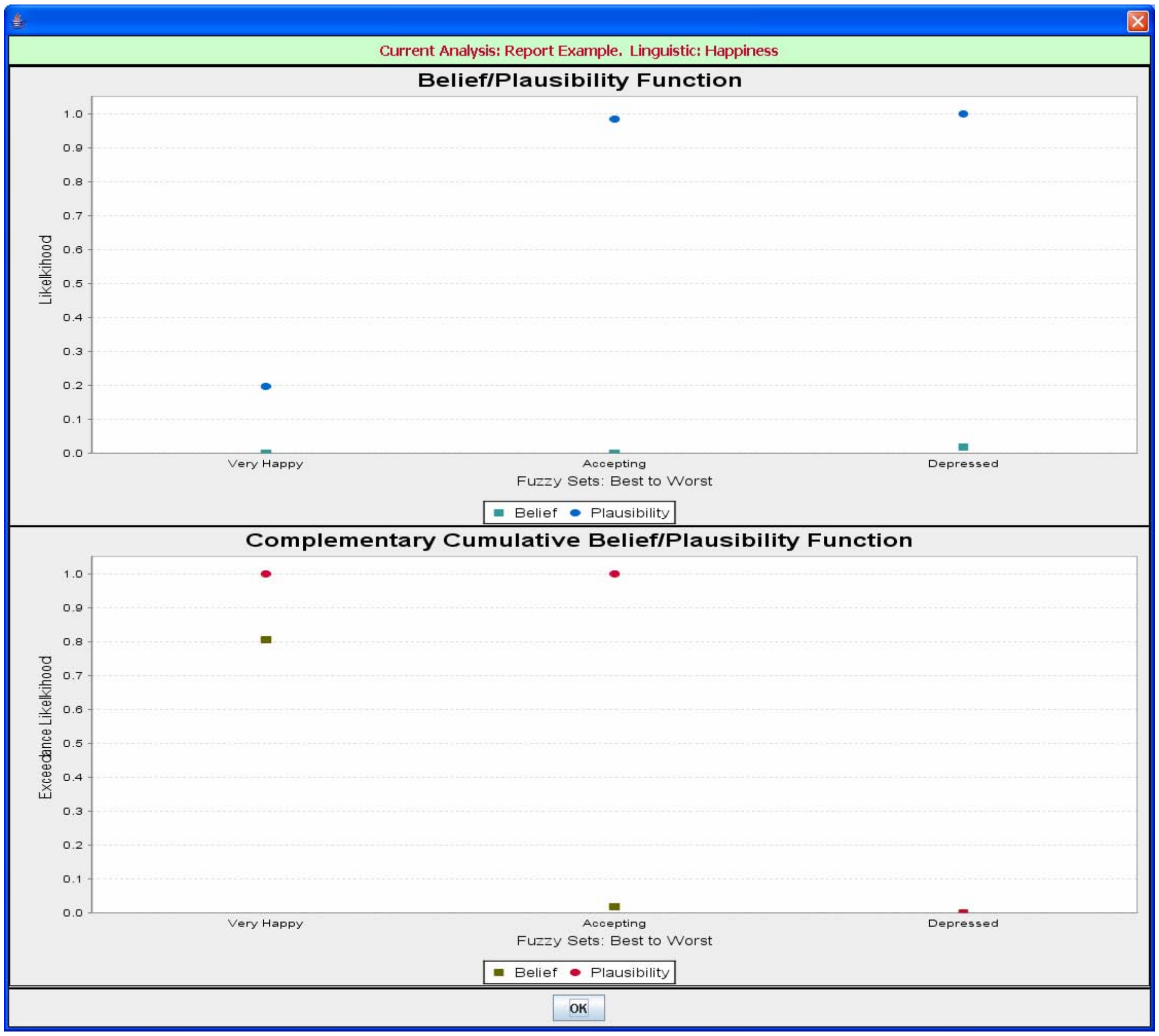
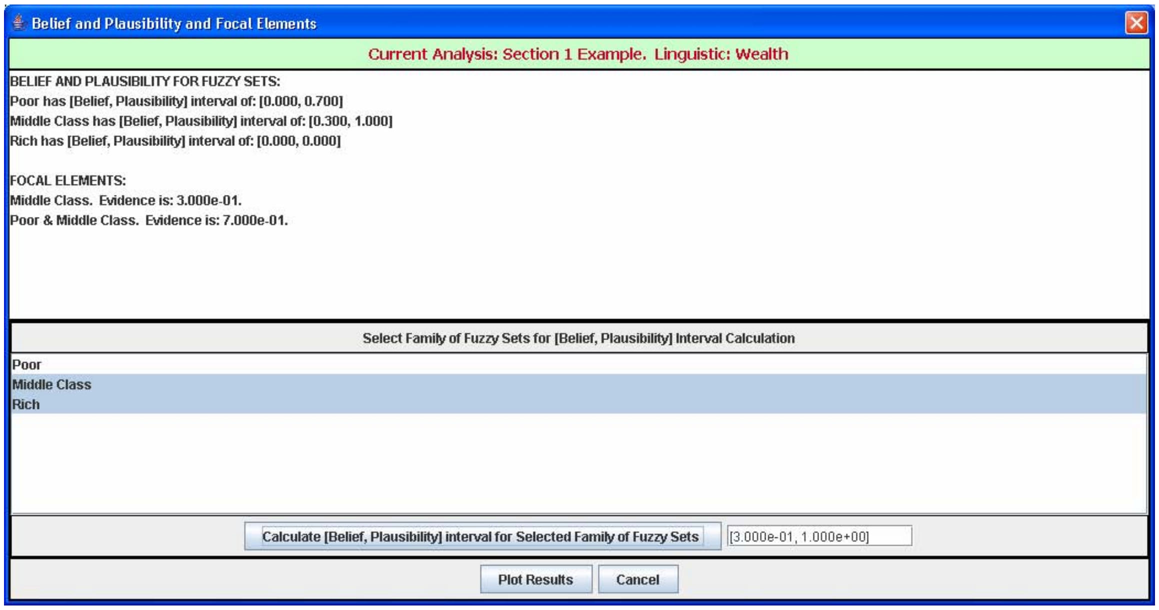


Figure 2-38. Graphical Results for "Happiness" for "John"

Results for any linguistic (basic or rule) can be generated. Figures 2-39 through 2-41 show the results for "Wealth."

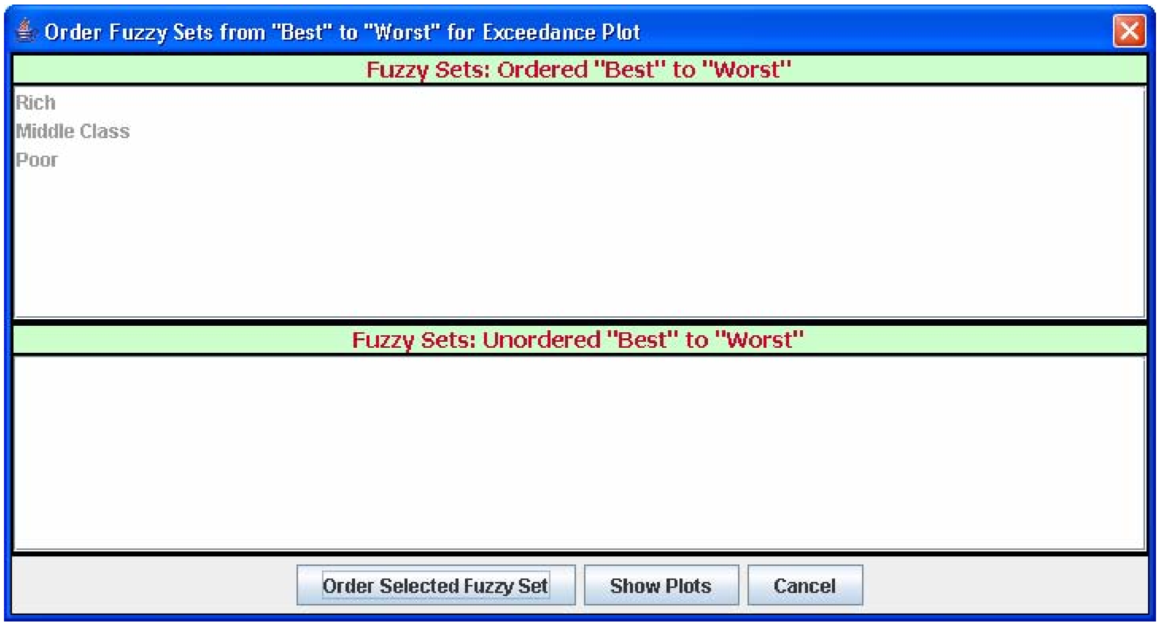

Belief and Plausibility and Focal Elements
Current Analysis: Section 1 Example. Linguistic: Wealth

```
BELIEF AND PLAUSIBILITY FOR FUZZY SETS:
Poor has [Belief, Plausibility] interval of: [0.000, 0.700]
Middle Class has [Belief, Plausibility] interval of: [0.300, 1.000]
Rich has [Belief, Plausibility] interval of: [0.000, 0.000]

FOCAL ELEMENTS:
Middle Class. Evidence is: 3.000e-01.
Poor & Middle Class. Evidence is: 7.000e-01.
```

Select Family of Fuzzy Sets for [Belief, Plausibility] Interval Calculation
Poor
Middle Class
Rich

Calculate [Belief, Plausibility] interval for Selected Family of Fuzzy Sets
[3.000e-01, 1.000e+00]

Figure 2-39. "Wealth" for "John"



Fuzzy Sets: Ordered "Best" to "Worst"
Rich
Middle Class
Poor

Fuzzy Sets: Unordered "Best" to "Worst"

Figure 2-40. "Wealth" Fuzzy Sets Ordered "Best" to "Worst"

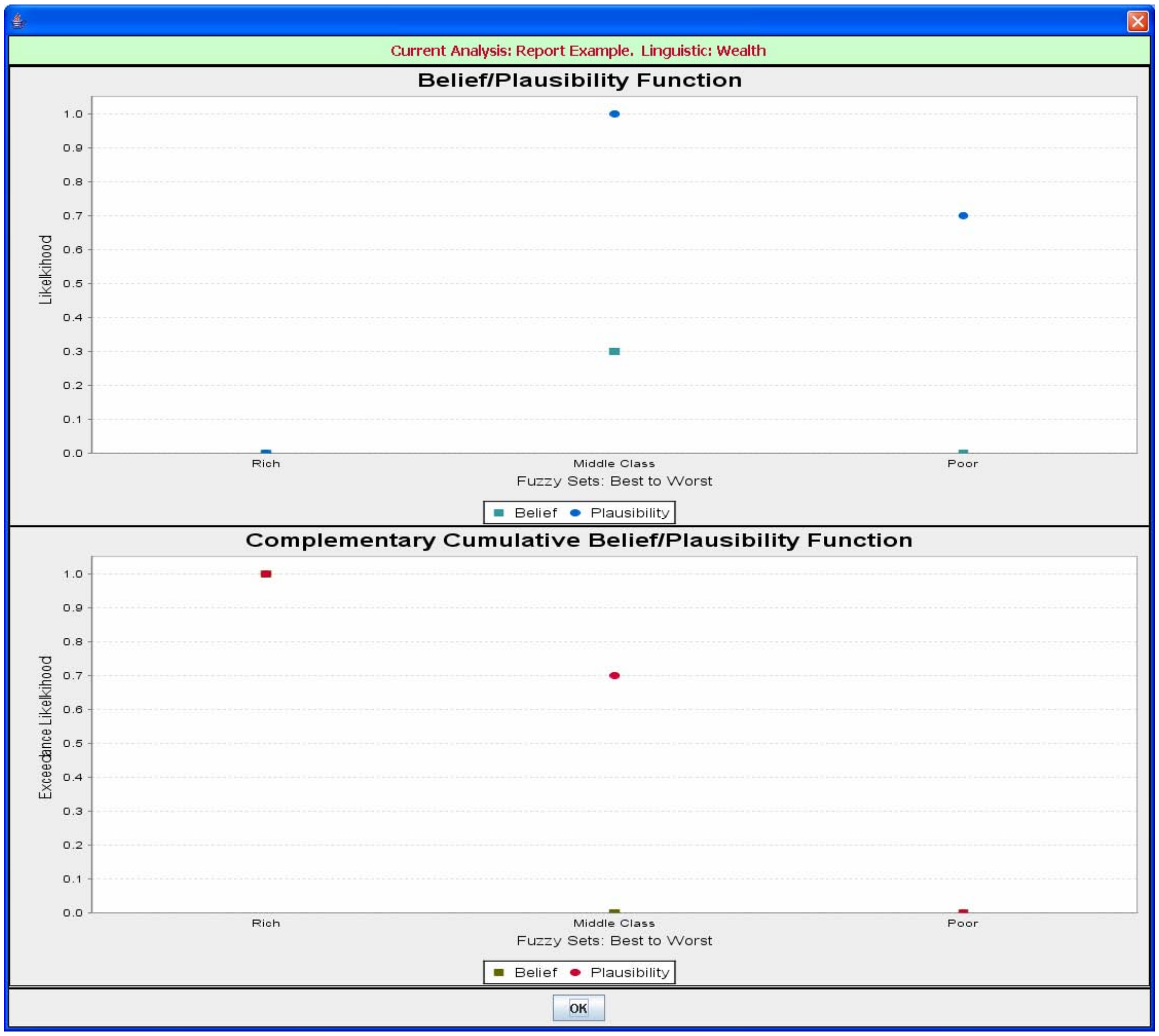

Current Analysis: Report Example. Linguistic: Wealth

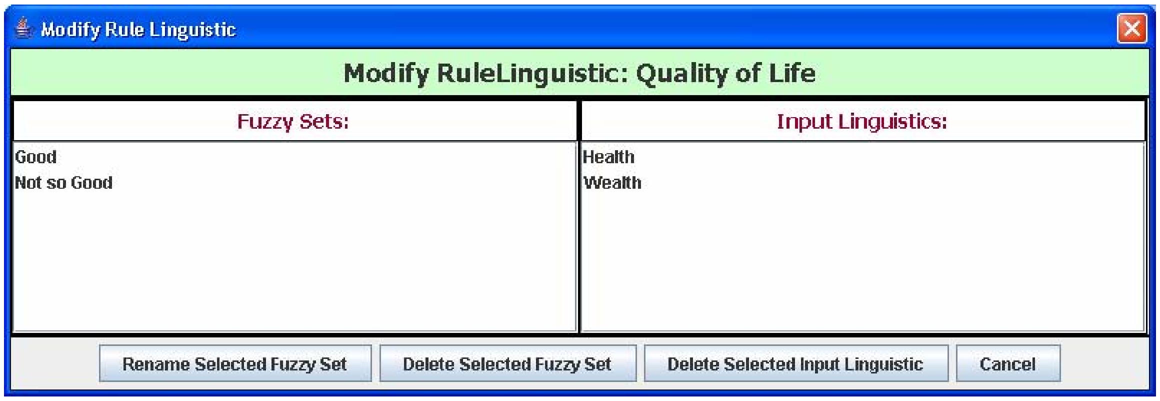
Figure 2-41. Graphical Results for “Wealth” for “John”

2.7 *Saving an Analysis*

The current analysis – "Section 1 Example" in this case – can be saved in xml format by clicking on the **Save** button on the frame button bar (shown, for example, on Figure 2-13). Using **Utilities|SaveSerialized**, the analysis can also be saved in Java serialized form as described in Appendix A.

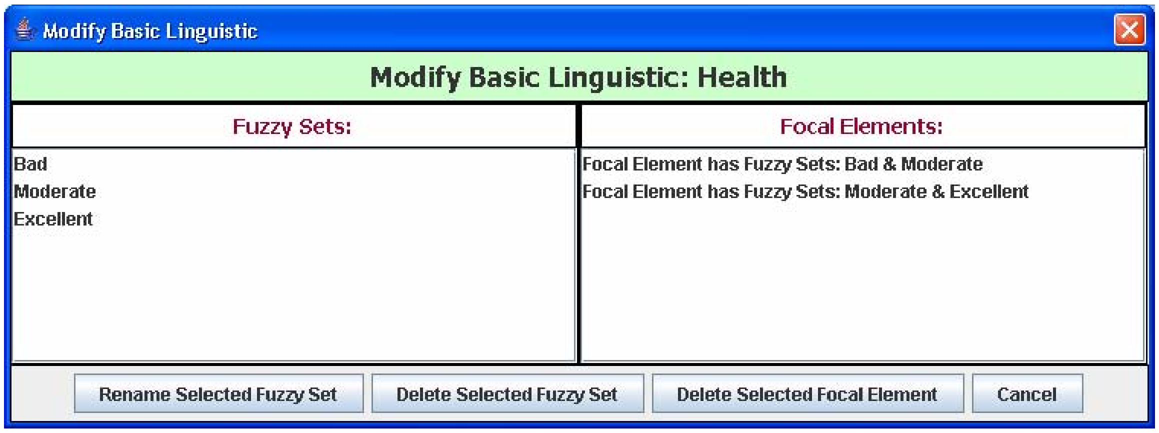
2.8 *Changing an Existing Analysis*

Changes to an existing analysis may be made. For example, fuzzy sets and input linguistics for a rule-based linguistic can be changed as indicated in Figure 2-42. Fuzzy sets and focal elements for a basic linguistic can be changed as indicated in Figure 2-43.


Modify Rule Linguistic
Modify RuleLinguistic: Quality of Life
Fuzzy Sets:
Input Linguistics:
Rename Selected Fuzzy Set
Delete Selected Fuzzy Set
Delete Selected Input Linguistic
Cancel

Fuzzy Sets:	Input Linguistics:
Good	Health
Not so Good	Wealth

Figure 2-42. Dialog for Changing Fuzzy Sets and Input Linguistics for a RuleLinguistic


Modify Basic Linguistic
Modify Basic Linguistic: Health
Fuzzy Sets:
Focal Elements:
Rename Selected Fuzzy Set
Delete Selected Fuzzy Set
Delete Selected Focal Element
Cancel

Fuzzy Sets:	Focal Elements:
Bad	Focal Element has Fuzzy Sets: Bad & Moderate
Moderate	Focal Element has Fuzzy Sets: Moderate & Excellent
Excellent	

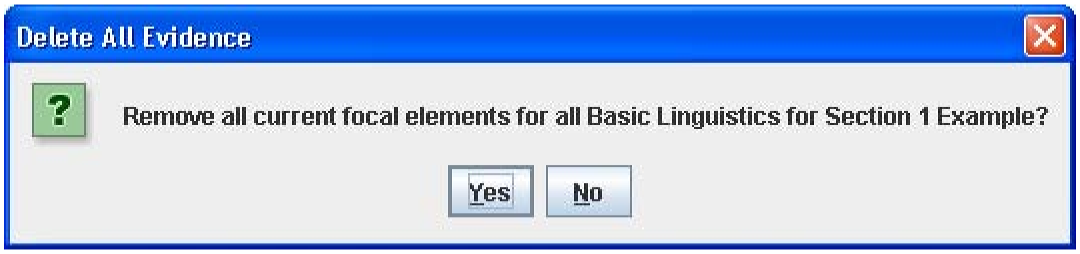
Figure 2-43. Dialog for Changing Fuzzy Sets and Focal Elements for a BasicLinguistic

2.9 Creating Another Specific Situation

To evaluate another situation, (e.g., another person such as "Mary" for our example), you can select **File|Save As** to create a new file for Mary, using the existing model for "John." Type a file name to represent the new person, "Mary," and then change the evidence for John to apply to Mary, as explained in this section.

To delete *all* focal elements from the current model

- Left-click the top node in the tree; for example, the "Current Analysis: Section 1 Example" node in Figure 2-1.
- Right-click the highlighted selection to display the dialog in Figure 2-44.
- Click on **Yes** to remove all focal elements.
- Click on **No** if you do not want to remove all focal elements. You have cloned John and you may wish to change only a few focal elements, as discussed in Section 2.8.


Delete All Evidence

Remove all current focal elements for all Basic Linguistics for Section 1 Example?

Figure 2-44. Dialog for Removing all Focal Elements

3 A Complex Example

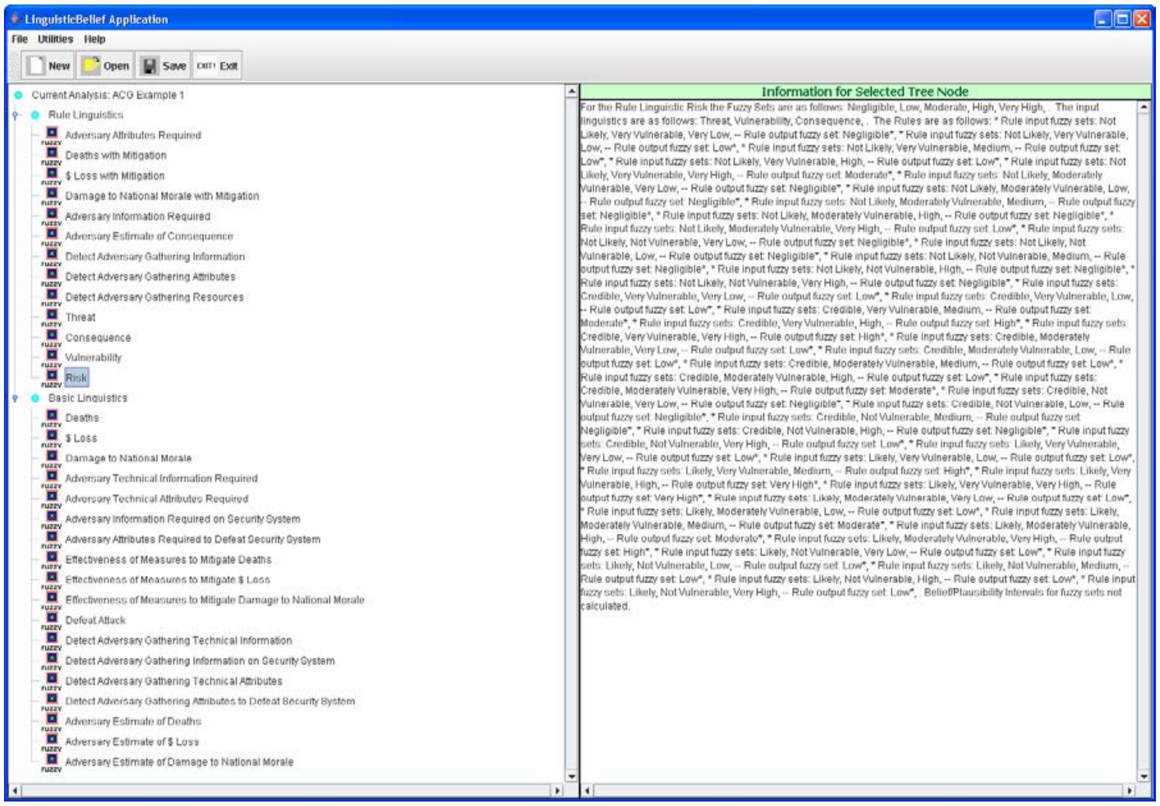
The prior sections used a simple example to illustrate the features of LinguisticBelief and to demonstrate how to use the tool.

This section briefly summarizes an application of LinguisticBelief for a real problem: evaluating and ranking the risk of terrorist scenarios.

3.1 An Example Model for Evaluating Terrorist Scenarios

Figure 3-1 shows the analysis model for this application. This model contains the linguistic variables and rule set for evaluating various terrorist scenarios as the Risk of the scenario.

Risk is a combination of Threat, Vulnerability, and Consequence. Threat is evaluated from the adversary perspective and includes the adversary perception of Vulnerability and Consequence. Specifically, Threat reasons on: Adversary Information Required, Adversary Attributes Required, and Adversary Estimate of Consequence. Vulnerability and Consequence are evaluated from the defender perspective. Vulnerability reasons on: Detect Adversary Gathering Resources and Defeat Attack. Consequence reasons on: Deaths, $ Loss, and Damage to National Morale.

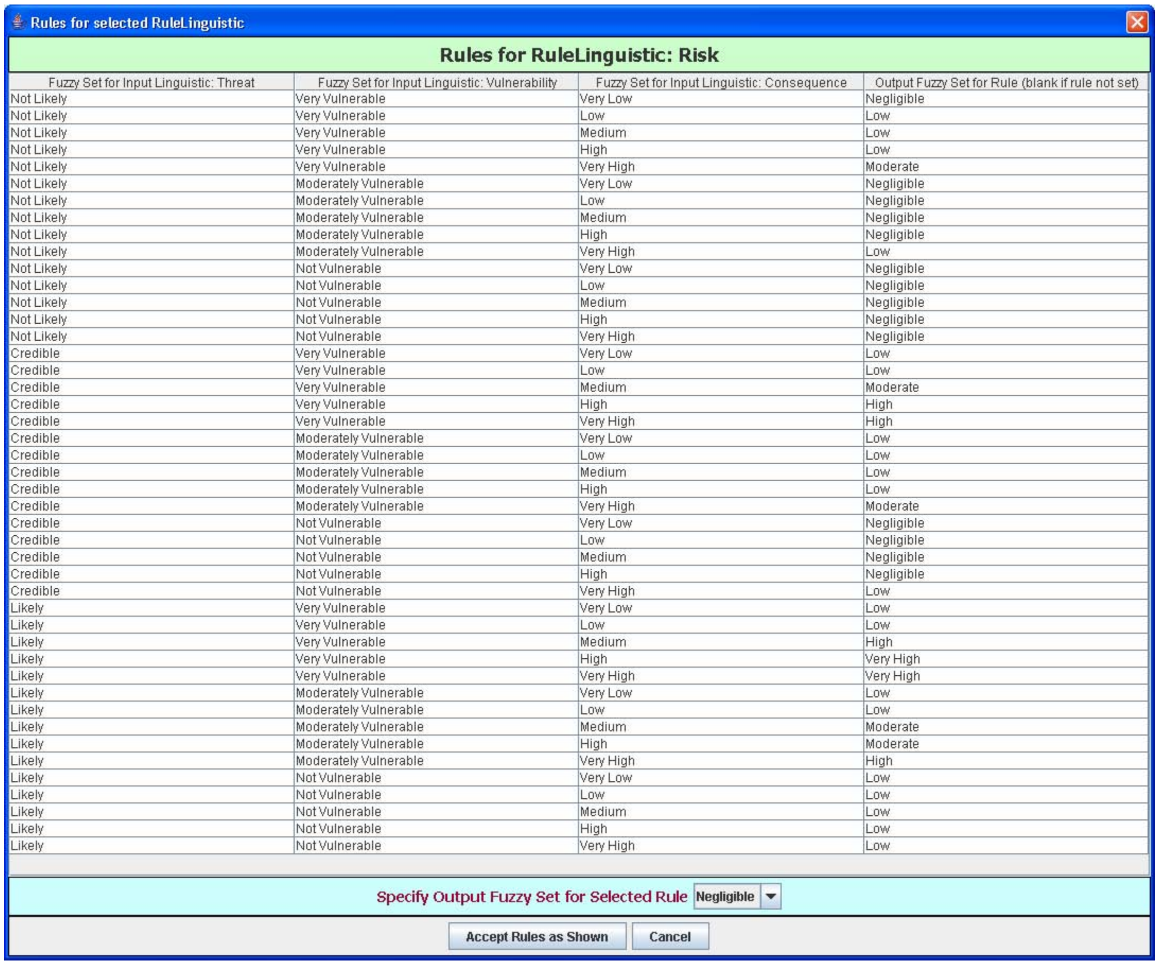

LinguisticBelief Application
File
Utilities
Help
New
Open
Save
EXIT Exit
Information for Selected Tree Node

- Current Analysis: ACG Example 1
- Rule Linguistics
  - Adversary Attributes Required
  - Deaths with Mitigation
  - $ Loss with Mitigation
  - Damage to National Morale with Mitigation
  - Adversary Information Required
  - Adversary Estimate of Consequence
  - Detect Adversary Gathering Information
  - Detect Adversary Gathering Attributes
  - Detect Adversary Gathering Resources
  - Threat
  - Consequence
  - Vulnerability
  - Risk
- Basic Linguistics
  - Deaths
  - $ Loss
  - Damage to National Morale
  - Adversary Technical Information Required
  - Adversary Technical Attributes Required
  - Adversary Information Required on Security System
  - Adversary Attributes Required to Defeat Security System
  - Effectiveness of Measures to Mitigate Deaths
  - Effectiveness of Measures to Mitigate $ Loss
  - Effectiveness of Measures to Mitigate Damage to National Morale
  - Defeat Attack
  - Detect Adversary Gathering Technical Information
  - Detect Adversary Gathering Information on Security System
  - Detect Adversary Gathering Technical Attributes
  - Detect Adversary Gathering Attributes to Defeat Security System
  - Adversary Estimate of Deaths
  - Adversary Estimate of $ Loss
  - Adversary Estimate of Damage to National Morale

For the Rule Linguistic Risk the Fuzzy Sets are as follows: Negligible, Low, Moderate, High, Very High, . The input linguistics are as follows: Threat, Vulnerability, Consequence, . The Rules are as follows: * Rule input fuzzy sets: Not Likely, Very Vulnerable, Very Low, -- Rule output fuzzy set: Negligible*, * Rule input fuzzy sets: Not Likely, Very Vulnerable, Low, -- Rule output fuzzy set: Low*, * Rule input fuzzy sets: Not Likely, Very Vulnerable, Medium, -- Rule output fuzzy set: Low*, * Rule input fuzzy sets: Not Likely, Very Vulnerable, High, -- Rule output fuzzy set: Low*, * Rule input fuzzy sets: Not Likely, Very Vulnerable, Very High, -- Rule output fuzzy set: Moderate*, * Rule input fuzzy sets: Not Likely, Moderately Vulnerable, Very Low, -- Rule output fuzzy set: Negligible*, * Rule input fuzzy sets: Not Likely, Moderately Vulnerable, Low, -- Rule output fuzzy set: Negligible*, * Rule input fuzzy sets: Not Likely, Moderately Vulnerable, Medium, -- Rule output fuzzy set: Negligible*, * Rule input fuzzy sets: Not Likely, Moderately Vulnerable, High, -- Rule output fuzzy set: Negligible*, * Rule input fuzzy sets: Not Likely, Moderately Vulnerable, Very High, -- Rule output fuzzy set: Low*, * Rule input fuzzy sets: Not Likely, Not Vulnerable, Very Low, -- Rule output fuzzy set: Negligible*, * Rule input fuzzy sets: Not Likely, Not Vulnerable, Low, -- Rule output fuzzy set: Negligible*, * Rule input fuzzy sets: Not Likely, Not Vulnerable, Medium, -- Rule output fuzzy set: Negligible*, * Rule input fuzzy sets: Not Likely, Not Vulnerable, High, -- Rule output fuzzy set: Negligible*, * Rule input fuzzy sets: Not Likely, Not Vulnerable, Very High, -- Rule output fuzzy set: Negligible*, * Rule input fuzzy sets: Credible, Very Vulnerable, Very Low, -- Rule output fuzzy set: Low*, * Rule input fuzzy sets: Credible, Very Vulnerable, Low, -- Rule output fuzzy set: Low*, * Rule input fuzzy sets: Credible, Very Vulnerable, Medium, -- Rule output fuzzy set: Moderate*, * Rule input fuzzy sets: Credible, Very Vulnerable, High, -- Rule output fuzzy set: High*, * Rule input fuzzy sets: Credible, Very Vulnerable, Very High, -- Rule output fuzzy set: High*, * Rule input fuzzy sets: Credible, Moderately Vulnerable, Very Low, -- Rule output fuzzy set: Low*, * Rule input fuzzy sets: Credible, Moderately Vulnerable, Low, -- Rule output fuzzy set: Low*, * Rule input fuzzy sets: Credible, Moderately Vulnerable, Medium, -- Rule output fuzzy set: Low*, * Rule input fuzzy sets: Credible, Moderately Vulnerable, High, -- Rule output fuzzy set: Low*, * Rule input fuzzy sets: Credible, Moderately Vulnerable, Very High, -- Rule output fuzzy set: Moderate*, * Rule input fuzzy sets: Credible, Not Vulnerable, Very Low, -- Rule output fuzzy set: Negligible*, * Rule input fuzzy sets: Credible, Not Vulnerable, Low, -- Rule output fuzzy set: Negligible*, * Rule input fuzzy sets: Credible, Not Vulnerable, Medium, -- Rule output fuzzy set: Negligible*, * Rule input fuzzy sets: Credible, Not Vulnerable, High, -- Rule output fuzzy set: Negligible*, * Rule input fuzzy sets: Credible, Not Vulnerable, Very High, -- Rule output fuzzy set: Low*, * Rule input fuzzy sets: Likely, Very Vulnerable, Very Low, -- Rule output fuzzy set: Low*, * Rule input fuzzy sets: Likely, Very Vulnerable, Low, -- Rule output fuzzy set: Low*, * Rule input fuzzy sets: Likely, Very Vulnerable, Medium, -- Rule output fuzzy set: High*, * Rule input fuzzy sets: Likely, Very Vulnerable, High, -- Rule output fuzzy set: Very High*, * Rule input fuzzy sets: Likely, Very Vulnerable, Very High, -- Rule output fuzzy set: Very High*, * Rule input fuzzy sets: Likely, Moderately Vulnerable, Very Low, -- Rule output fuzzy set: Low*, * Rule input fuzzy sets: Likely, Moderately Vulnerable, Low, -- Rule output fuzzy set: Low*, * Rule input fuzzy sets: Likely, Moderately Vulnerable, Medium, -- Rule output fuzzy set: Moderate*, * Rule input fuzzy sets: Likely, Moderately Vulnerable, High, -- Rule output fuzzy set: Moderate*, * Rule input fuzzy sets: Likely, Moderately Vulnerable, Very High, -- Rule output fuzzy set: High*, * Rule input fuzzy sets: Likely, Not Vulnerable, Very Low, -- Rule output fuzzy set: Low*, * Rule input fuzzy sets: Likely, Not Vulnerable, Low, -- Rule output fuzzy set: Low*, * Rule input fuzzy sets: Likely, Not Vulnerable, Medium, -- Rule output fuzzy set: Low*, * Rule input fuzzy sets: Likely, Not Vulnerable, High, -- Rule output fuzzy set: Low*, * Rule input fuzzy sets: Likely, Not Vulnerable, Very High, -- Rule output fuzzy set: Low*, . Belief/Plausibility Intervals for fuzzy sets not calculated.

Figure 3-1. Analysis Model for Risk of a Scenario

Figure 3-2 shows the rule base created for the top Rule Linguistic “Risk.”



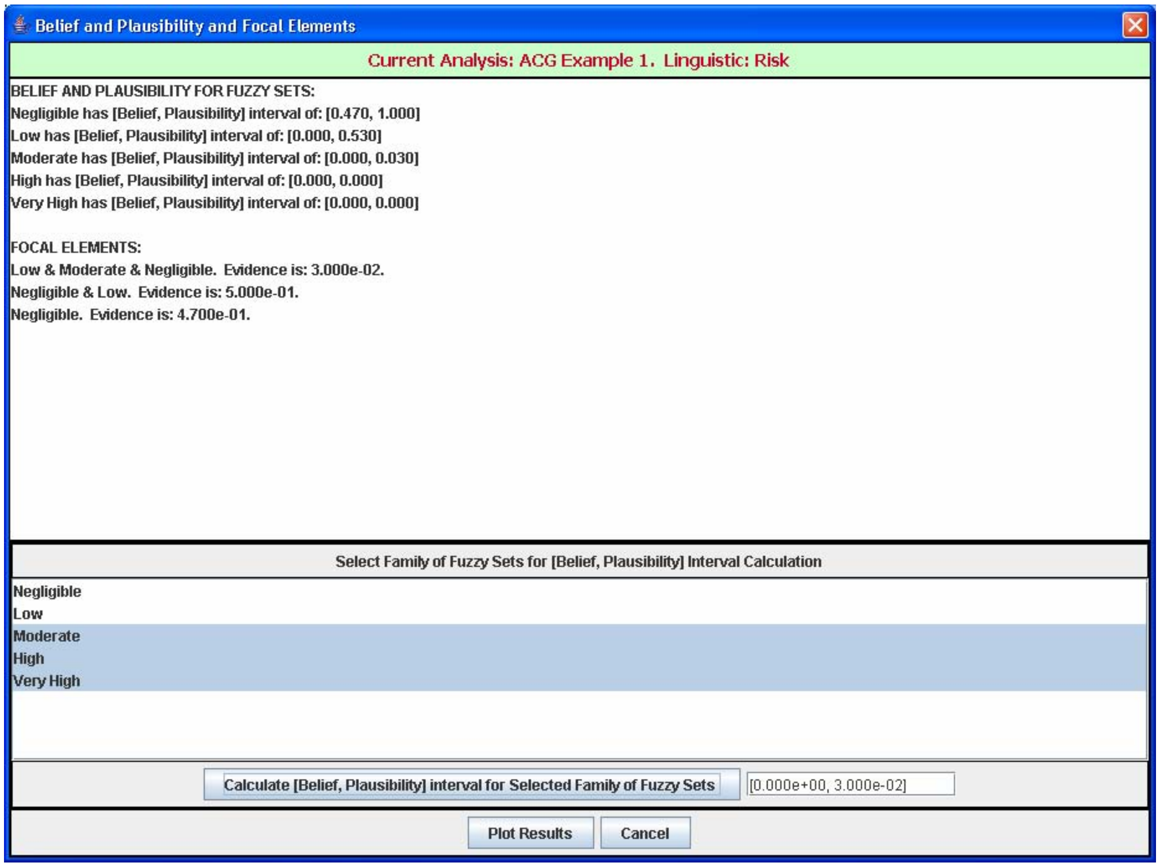
Rules for RuleLinguistic: Risk

Fuzzy Set for Input Linguistic: Threat	Fuzzy Set for Input Linguistic: Vulnerability	Fuzzy Set for Input Linguistic: Consequence	Output Fuzzy Set for Rule (blank if rule not set)
Not Likely	Very Vulnerable	Very Low	Negligible
Not Likely	Very Vulnerable	Low	Low
Not Likely	Very Vulnerable	Medium	Low
Not Likely	Very Vulnerable	High	Low
Not Likely	Very Vulnerable	Very High	Moderate
Not Likely	Moderately Vulnerable	Very Low	Negligible
Not Likely	Moderately Vulnerable	Low	Negligible
Not Likely	Moderately Vulnerable	Medium	Negligible
Not Likely	Moderately Vulnerable	High	Negligible
Not Likely	Moderately Vulnerable	Very High	Low
Not Likely	Not Vulnerable	Very Low	Negligible
Not Likely	Not Vulnerable	Low	Negligible
Not Likely	Not Vulnerable	Medium	Negligible
Not Likely	Not Vulnerable	High	Negligible
Not Likely	Not Vulnerable	Very High	Negligible
Credible	Very Vulnerable	Very Low	Low
Credible	Very Vulnerable	Low	Low
Credible	Very Vulnerable	Medium	Moderate
Credible	Very Vulnerable	High	High
Credible	Very Vulnerable	Very High	High
Credible	Moderately Vulnerable	Very Low	Low
Credible	Moderately Vulnerable	Low	Low
Credible	Moderately Vulnerable	Medium	Low
Credible	Moderately Vulnerable	High	Low
Credible	Moderately Vulnerable	Very High	Moderate
Credible	Not Vulnerable	Very Low	Negligible
Credible	Not Vulnerable	Low	Negligible
Credible	Not Vulnerable	Medium	Negligible
Credible	Not Vulnerable	High	Negligible
Credible	Not Vulnerable	Very High	Low
Likely	Very Vulnerable	Very Low	Low
Likely	Very Vulnerable	Low	Low
Likely	Very Vulnerable	Medium	High
Likely	Very Vulnerable	High	Very High
Likely	Very Vulnerable	Very High	Very High
Likely	Moderately Vulnerable	Very Low	Low
Likely	Moderately Vulnerable	Low	Low
Likely	Moderately Vulnerable	Medium	Moderate
Likely	Moderately Vulnerable	High	Moderate
Likely	Moderately Vulnerable	Very High	High
Likely	Not Vulnerable	Very Low	Low
Likely	Not Vulnerable	Low	Low
Likely	Not Vulnerable	Medium	Low
Likely	Not Vulnerable	High	Low
Likely	Not Vulnerable	Very High	Low

Specify Output Fuzzy Set for Selected Rule: Negligible

Figure 3-2. Rules for Risk

For *one* scenario of concern, Figure 3-3 shows the evaluation of Risk using dummy data for the focal elements for that scenario.


Belief and Plausibility and Focal Elements
Current Analysis: ACG Example 1. Linguistic: Risk

```
BELIEF AND PLAUSIBILITY FOR FUZZY SETS:
Negligible has [Belief, Plausibility] interval of: [0.470, 1.000]
Low has [Belief, Plausibility] interval of: [0.000, 0.530]
Moderate has [Belief, Plausibility] interval of: [0.000, 0.030]
High has [Belief, Plausibility] interval of: [0.000, 0.000]
Very High has [Belief, Plausibility] interval of: [0.000, 0.000]

FOCAL ELEMENTS:
Low & Moderate & Negligible.  Evidence is: 3.000e-02.
Negligible & Low.  Evidence is: 5.000e-01.
Negligible.  Evidence is: 4.700e-01.
```

Select Family of Fuzzy Sets for [Belief, Plausibility] Interval Calculation
Negligible
Low
Moderate
High
Very High
Calculate [Belief, Plausibility] interval for Selected Family of Fuzzy Sets
[0.000e+00, 3.000e-02]
Plot Results
Cancel

Figure 3-3. Results for Risk for Scenario 1

The graphical results for Risk for Scenario 1 are shown in Figure 3-4.

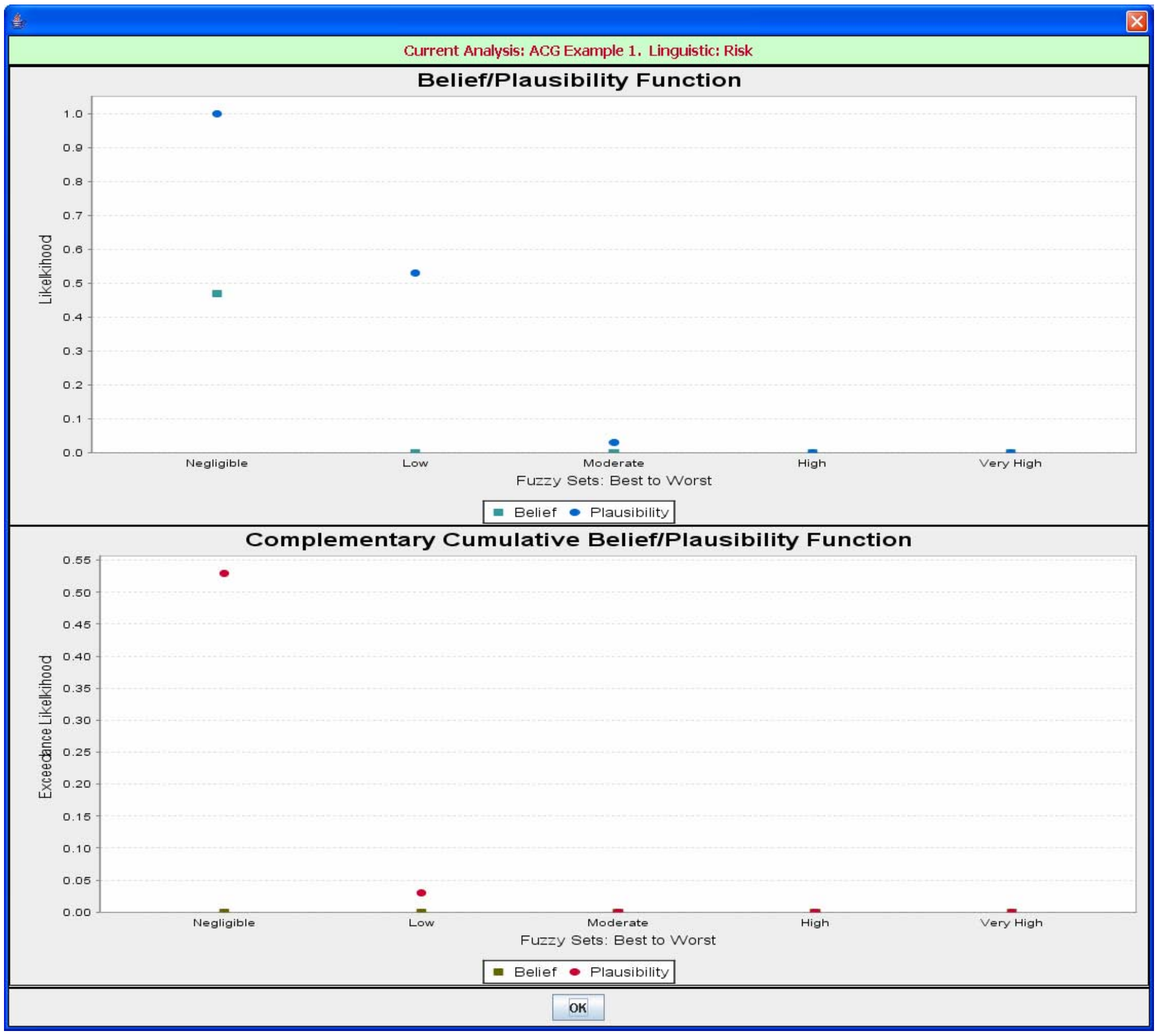
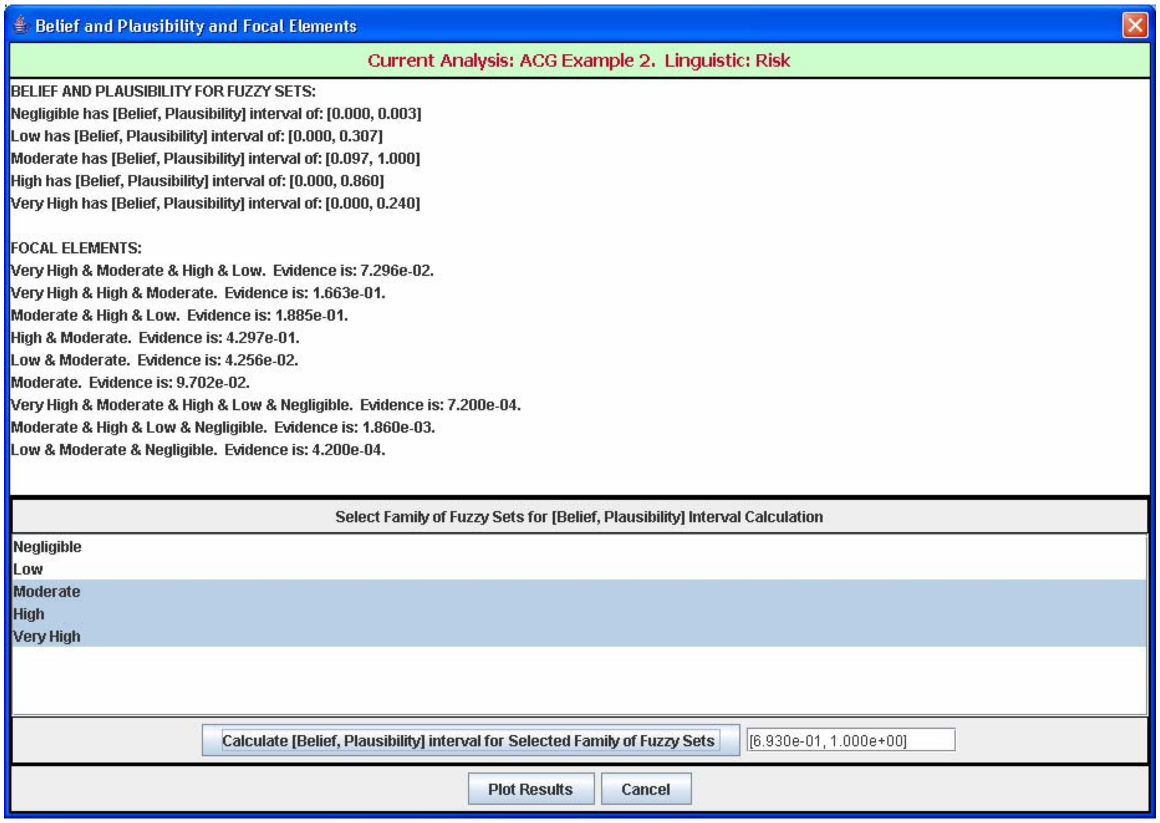

Current Analysis: ACG Example 1. Linguistic: Risk

Figure 3-4. Graphical Results for Risk for Scenario 1

For *another* scenario of concern, Figure 3-5 shows the evaluation of Risk using dummy data for the focal elements for that scenario. Figure 3-6 shows the graphical results for Scenario 2.


Belief and Plausibility and Focal Elements
Current Analysis: ACG Example 2. Linguistic: Risk

```
BELIEF AND PLAUSIBILITY FOR FUZZY SETS:
Negligible has [Belief, Plausibility] interval of: [0.000, 0.003]
Low has [Belief, Plausibility] interval of: [0.000, 0.307]
Moderate has [Belief, Plausibility] interval of: [0.097, 1.000]
High has [Belief, Plausibility] interval of: [0.000, 0.860]
Very High has [Belief, Plausibility] interval of: [0.000, 0.240]

FOCAL ELEMENTS:
Very High & Moderate & High & Low.  Evidence is: 7.296e-02.
Very High & High & Moderate.  Evidence is: 1.663e-01.
Moderate & High & Low.  Evidence is: 1.885e-01.
High & Moderate.  Evidence is: 4.297e-01.
Low & Moderate.  Evidence is: 4.256e-02.
Moderate.  Evidence is: 9.702e-02.
Very High & Moderate & High & Low & Negligible.  Evidence is: 7.200e-04.
Moderate & High & Low & Negligible.  Evidence is: 1.860e-03.
Low & Moderate & Negligible.  Evidence is: 4.200e-04.
```

Select Family of Fuzzy Sets for [Belief, Plausibility] Interval Calculation
Negligible
Low
Moderate
High
Very High
Calculate [Belief, Plausibility] interval for Selected Family of Fuzzy Sets
[6.930e-01, 1.000e+00]
Plot Results
Cancel

Figure 3-5. Results for Risk for Scenario 2

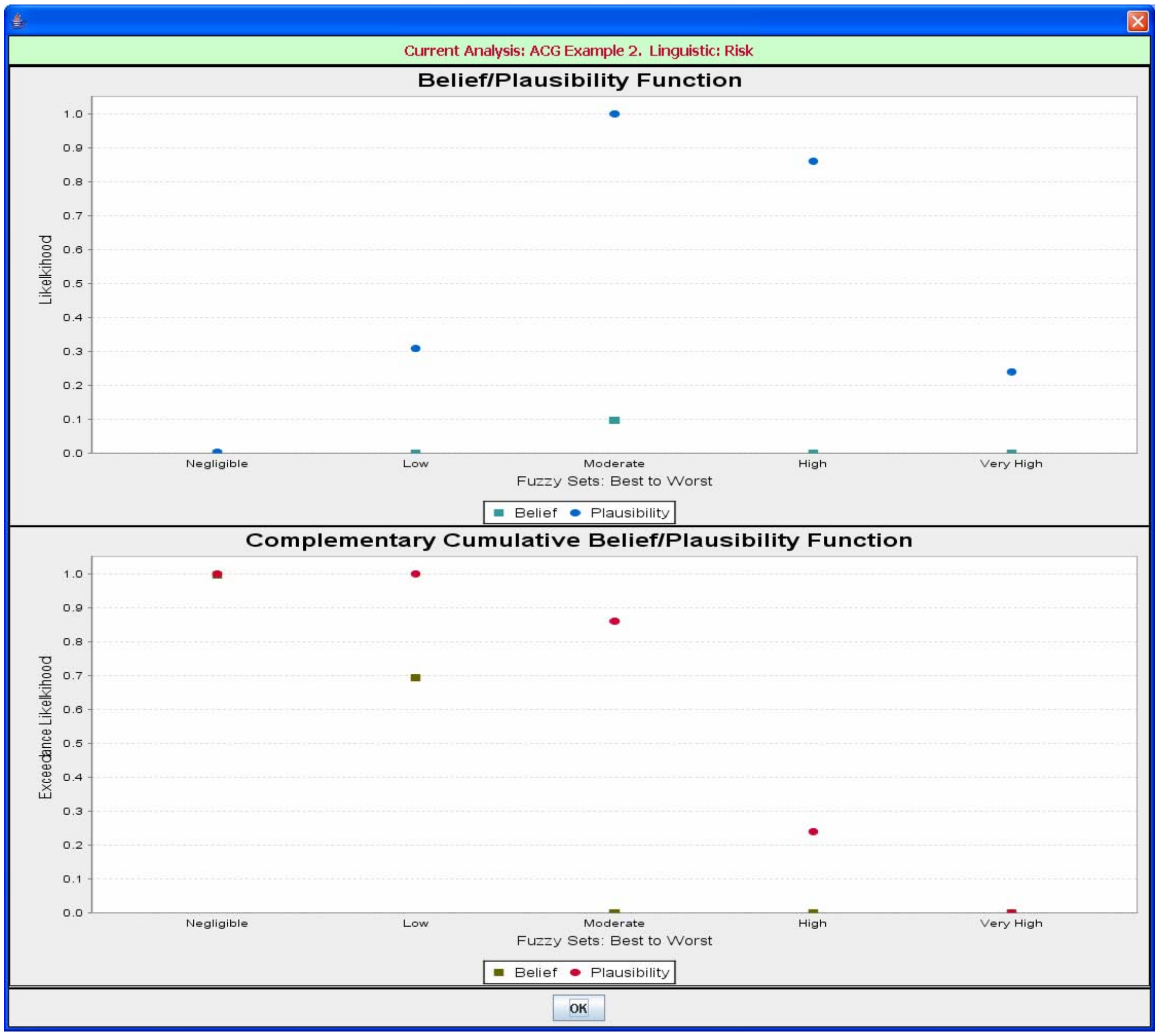


Figure 3-6. Graphical Results for Risk for Scenario 2

3.2 Ranking Scenarios

For two scenarios, the graphs in the lower panels of Figures 3-4 and 3-6 summarize risk as a “likelihood” of exceedance of fuzzy sets, with the fuzzy sets ranked from least to most risk. These results can be used to rank the scenarios.

Let the fuzzy set f_a be “worse” than the fuzzy set f_b. For any specific scenario “k:”

$$Plaus_{k\ exceed\ f_a} \leq Plaus_{k\ exceed\ f_b} \qquad \text{(Eqn. 1)}$$

where $Plaus_{k\ exceed\ f_a}$ is the plausibility of exceedance of fuzzy set f_a for scenario “k.”

Scenarios are ranked by non-zero plausibility of exceeding the “worst” given fuzzy set (decreasing). For scenarios with equal ranking by plausibility, these scenarios are subranked by belief of exceeding the fuzzy set (decreasing). Note that the ranking of the scenarios can be different for f_a than for f_b.

Ranking of a few scenarios can be easily done manually; however, for a large number of scenarios the ranking must be automated. The ranking process has been implemented in a Java utility code, RankScenarios, written by the author. This code iterates over all fuzzy sets, worst to best, and for each fuzzy set ranks the scenarios by non-zero decreasing plausibility of exceeding that fuzzy set, and – for scenarios with the same plausibility – subranks by decreasing belief of exceeding that fuzzy set.

For the two scenarios summarized in Figures 3-4 and 3-6, the ranking results from the RankScenarios code are as given in Listing 3-1.

Listing 3-1. Results from RankScenarios for Two Terrorist Scenarios

```
RANKING FOR SCENARIOS 1 and 2

For Exceeding Fuzzy Set High the Scenarios rank ordered (decreasing) are:
Scenario 2 has plausibility of exceedance of 0.24 and belief of exceedance of 0.0

For Exceeding Fuzzy Set Moderate the Scenarios rank ordered (decreasing) (not already ranked for a
worse fuzzy set) are:
        (null)

For Exceeding Fuzzy Set Low the Scenarios rank ordered (decreasing) (not already ranked for a worse
fuzzy set) are:
Scenario 1 has plausibility of exceedance of 0.03 and belief of exceedance of 0.0

For Exceeding Fuzzy Set Negligible the Scenarios rank ordered (decreasing) (not already ranked for a
worse fuzzy set) are:
        (null)
```

Figures 3-7 and 3-8 graphically display the ranking of these two scenarios. This type of graphical summary was proposed by John Cummings of SNL. Note that in these figures the fuzzy set name has been offset from the belief/plausibility interval for **exceeding** that fuzzy set to emphasize that these are exceedance plots. For example, in Figure 3-7, the belief/plausibility of exceeding "Negligible" Risk is 0/0.53, which is the belief/plausibility that Risk is either "Low," "Moderate," "High," or "Very High."

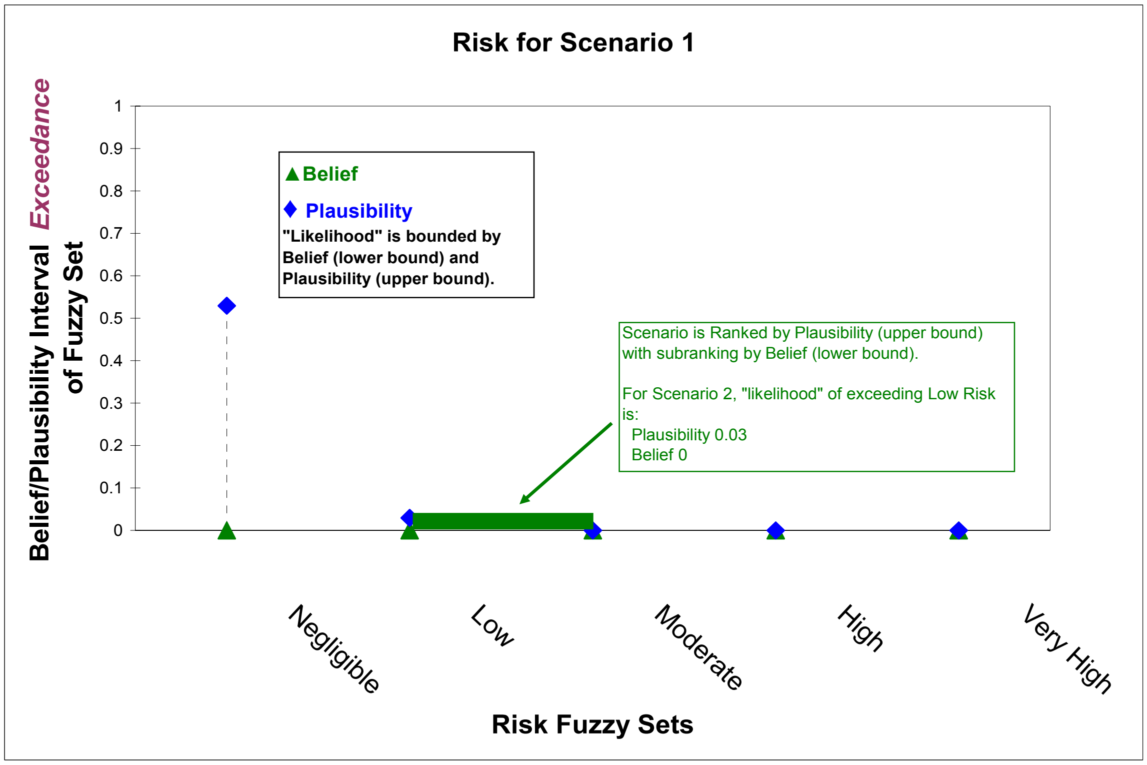


Figure 3-7. Graphical Presentation for Ranking of Scenario 1

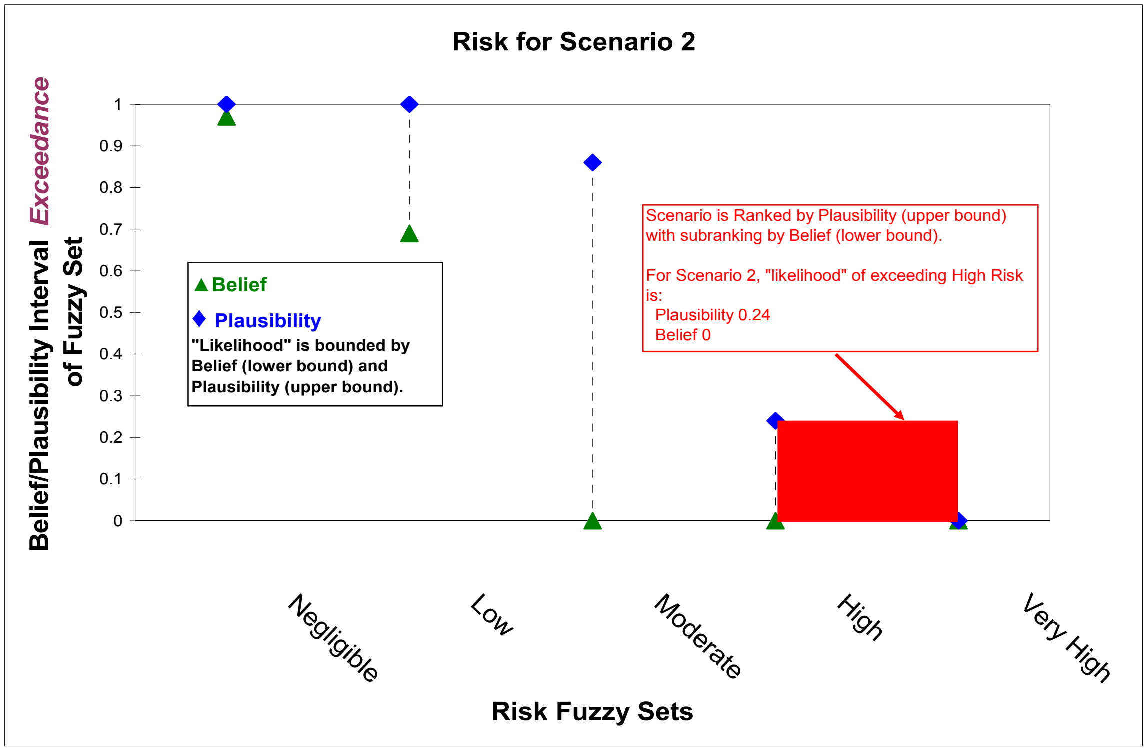


Figure 3-8. Graphical Presentation for Ranking of Scenario 2

The ranking technique presented in this section is not restricted to ranking terrorist scenarios. It can be used to rank results for any situation where the fuzzy sets for a variable of concern are ordered from “best” to “worst.” For example, Section 1 evaluated the “Happiness” of two individuals, “John” and “Janet,” and concluded that “John” is not as happy as “”Janet.” This ranking can be formalized using the RiskScenarios code with the results given in Listing 3-2.

Listing 3-2. Results from RankScenarios for Happiness

```
RANKING FOR JOHN AND JANET

For Exceeding Fuzzy Set Accepting the Scenarios rank ordered (decreasing)  (not already ranked for a
worse fuzzy set) are:
Scenario John has plausibility of exceedance of 1.0 and belief of exceedance of 0.016
Scenario Janet has plausibility of exceedance of 0.15 and belief of exceedance of 0.041

For Exceeding Fuzzy Set Very Happy the Scenarios rank ordered (decreasing)  (not already ranked for a
worse fuzzy set) are:
          (null)
```

Figures 3-9 and 3-10 show these rankings graphically.

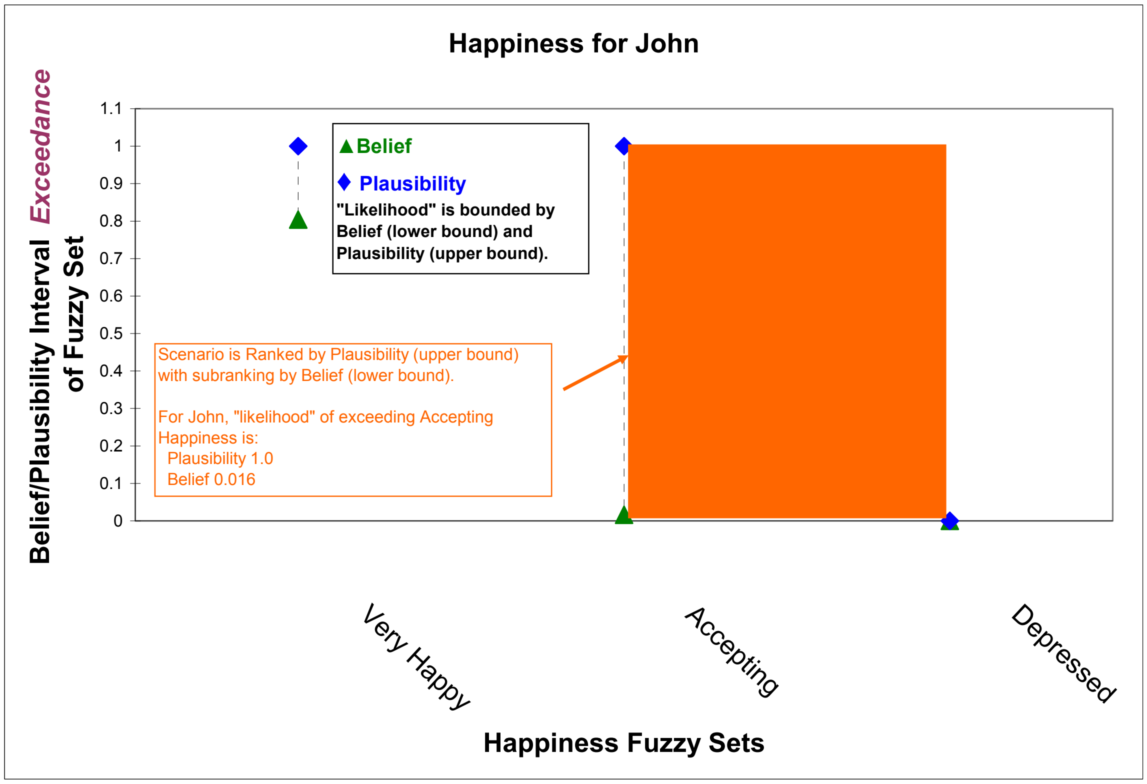


Figure 3-9. Graphical Presentation for Ranking of Happiness of John

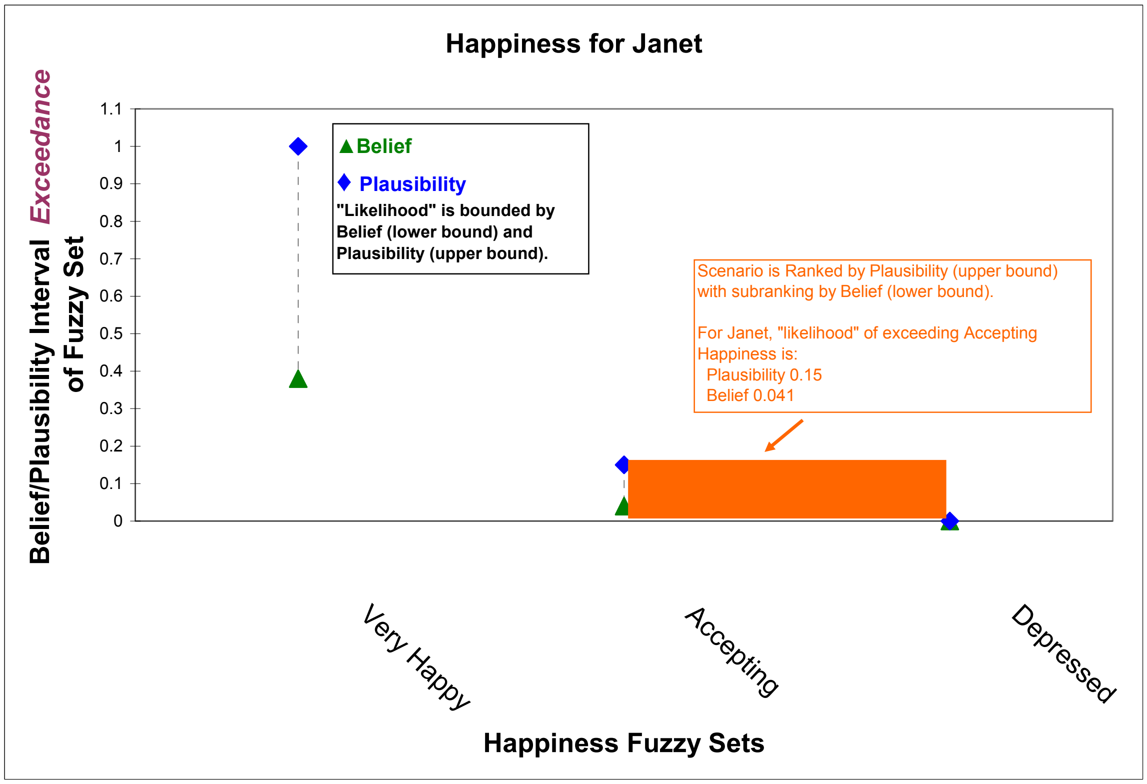


Figure 3-10. Graphical Presentation for Ranking of Happiness of Janet

4 Special Cases

This section illustrates two degenerate cases of belief/plausibility: probability and certainty. Also, the case with total ignorance is discussed.

All the examples in the earlier sections required use of the belief/plausibility measure of uncertainty because the evidence was non-specific.[8] For example, for the "Health" of "John" we assigned the following focal elements in the example in Section 1:

0.8 to {"Bad," "Moderate"}, and
0.2 to {"Moderate," "Excellent"}.

Using this body of evidence, the results for "Health" for "John" are given in Figure 4-1.

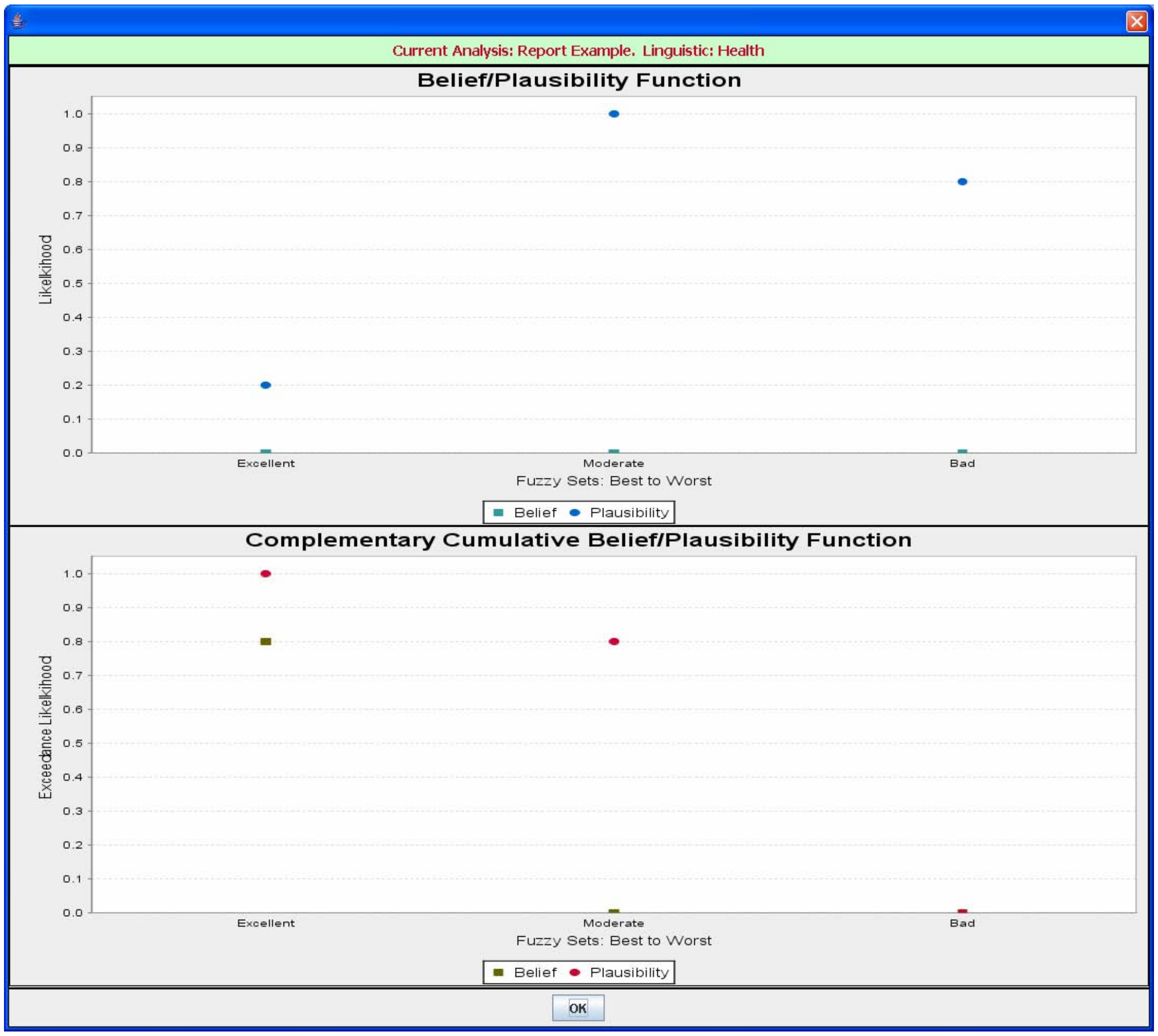


Figure 4-1. Graphical Results for "Health" for "John"

8 The focal elements were not all singletons. [Darby]

Note that the belief or plausibility for *exceedance* of a fuzzy set *cannot* in general be directly calculated from the belief or plausibility values for the fuzzy sets that are worse than the fuzzy set of concern. For example, the plausibility of exceedance of "Excellent" (1.0 as shown in the lower graph of Figure 4-1) is not the sum of the plausibility of "Moderate" and "Bad" (1.0 + 0.8 as shown in the upper graph Figure 4-1). As discussed in a reference, belief and plausibility must be calculated using the focal elements. [Darby] Specifically, the belief (Bel) and plausibility (Pl) for any event A are:

$$\begin{aligned} Bel(A) &= \sum_{B|B\subseteq A} m(B) \\ Pl(A) &= \sum_{B|A\cap B\neq 0} m(B) \end{aligned} \qquad \text{(Eqn. 2)}$$

where m(B) is the evidence assigned to event B (B is a focal element).

For the special case of probability, the probability of exceedance of a fuzzy set can be calculated directly as the sum of the probabilities for the fuzzy sets that are worse than the fuzzy set of concern. The next section provides an example where belief and plausibility are both probability.

4.1 Probability

If the evidence is specific enough, the focal elements are singletons and both belief/plausibility reduce to probability. That is, probability is a special case, or a degenerate case, of belief/plausibility.[9] For example, assume that for the "Health" of "Fred" we assign the following focal elements:

0.1 to {"Bad,"},
0.6 to {"Moderate,"}, and
0.3 to {"Excellent"}.

[9] Necessity/possibility is another special case of belief/plausibility. [Darby] In this situation the focal elements are nested.

Using this body of evidence, the results for “Health” for “Fred” are given in Figure 4-2.

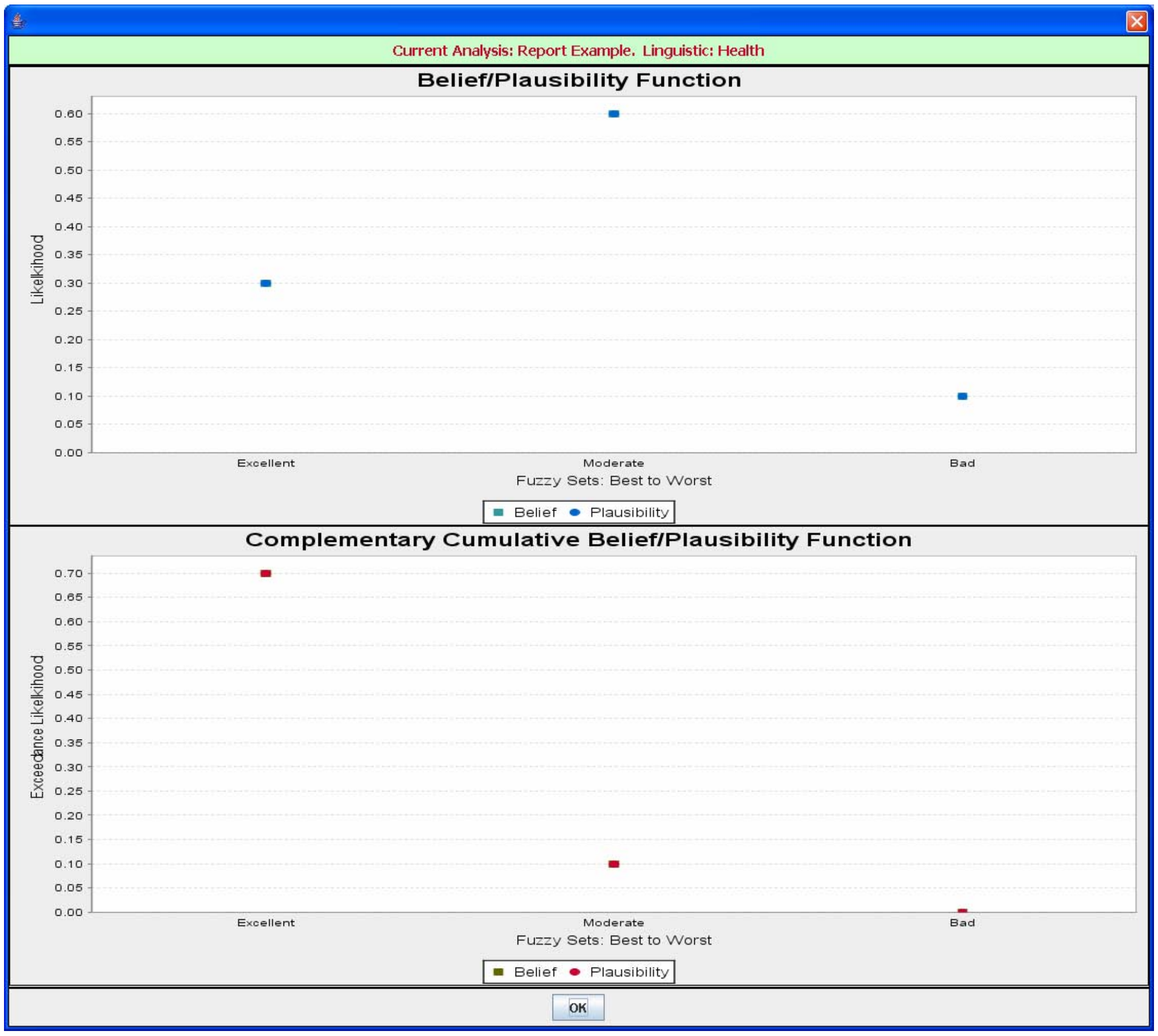


Figure 4-2. Graphical Results for “Health” for “Fred”

Note, in Figure 4-2 belief and plausibility both are a single value, which is the probability. Here, the belief/plausibility distribution is a probability distribution.

Note that the probability for exceedance of a fuzzy set *can* be directly calculated from the probability values for the fuzzy sets that are worse than the fuzzy set of concern. For example, the probability of exceedance of “Excellent” (0.7 as shown in the lower graph of Figure 4-2) is the sum of the probability of “Moderate” and “Bad” (0.6 + 0.1 as shown in the upper graph of Figure 4-2).

For probability, the focal elements are singletons; each focal element has a single fuzzy set. In Equation 2, if each B is a singleton, then Bel(A) and Pl(A) are the same: the Probability(A).

4.2 Certainty

Certainty is a special, or degenerate, case of probability. For example, assume we know the "Health" for "Richard" is "Moderate" based on information from his physician. For "Richard" we assign the following focal elements:

1.0 to {"Moderate"}.

Using this body of evidence, the results for "Health" for "Richard" are given in Figure 4-3.

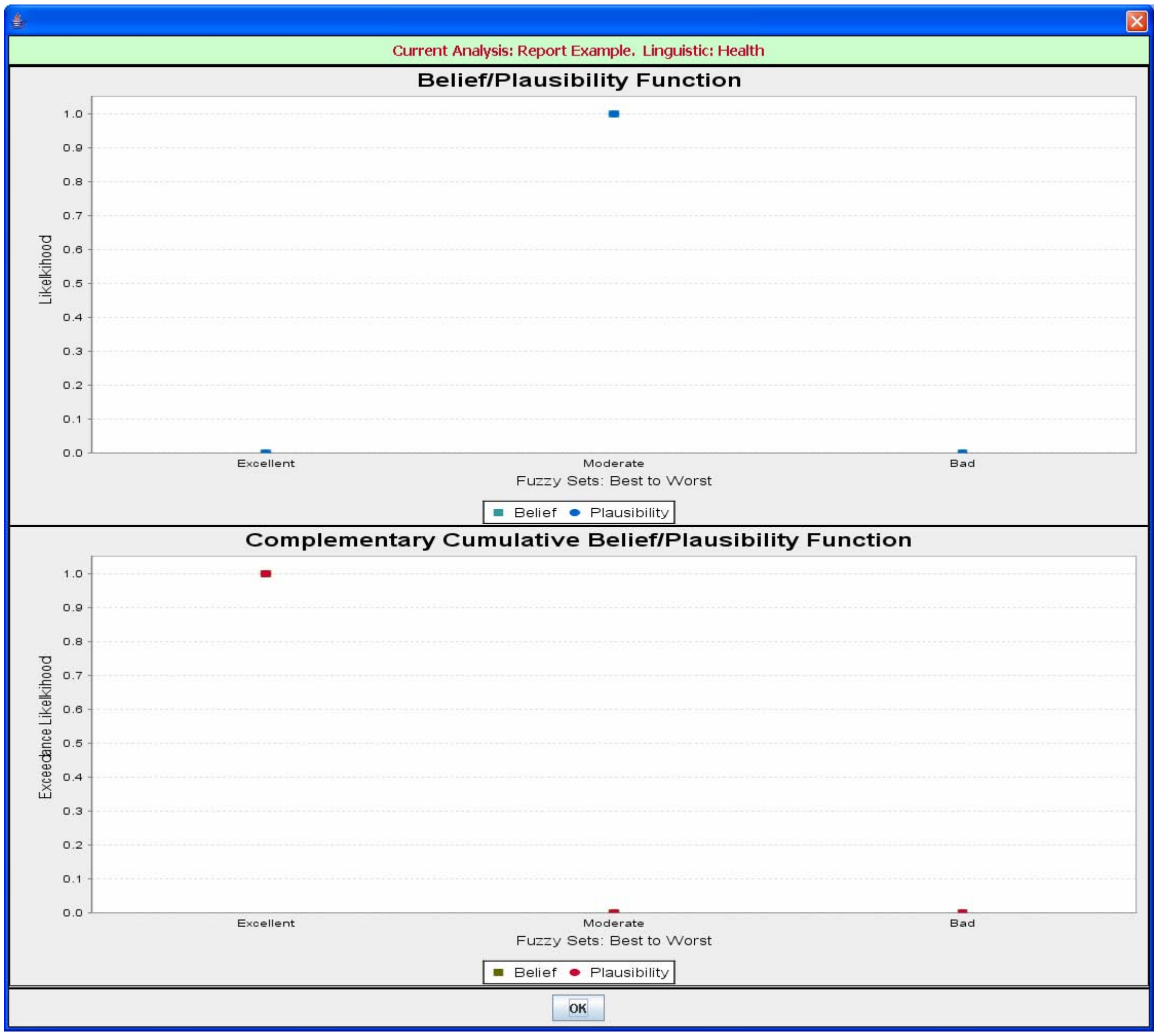


Figure 4-3. Graphical Results for "Health" for "Richard"

For "Richard," belief and plausibility both are a single value, probability, and we are certain that "Richard" has "Moderate" "Health," so the probability of "Moderate" is 1.0.

4.3 Total Ignorance

If we have no information as to the "Health" of an individual, say "Mary," we have total ignorance and the focal elements are as follows:

1.0 to {"Bad," "Moderate," "Excellent"}.

Using this body of evidence, the results for "Health" for "Mary" are given in Figure 4-4.

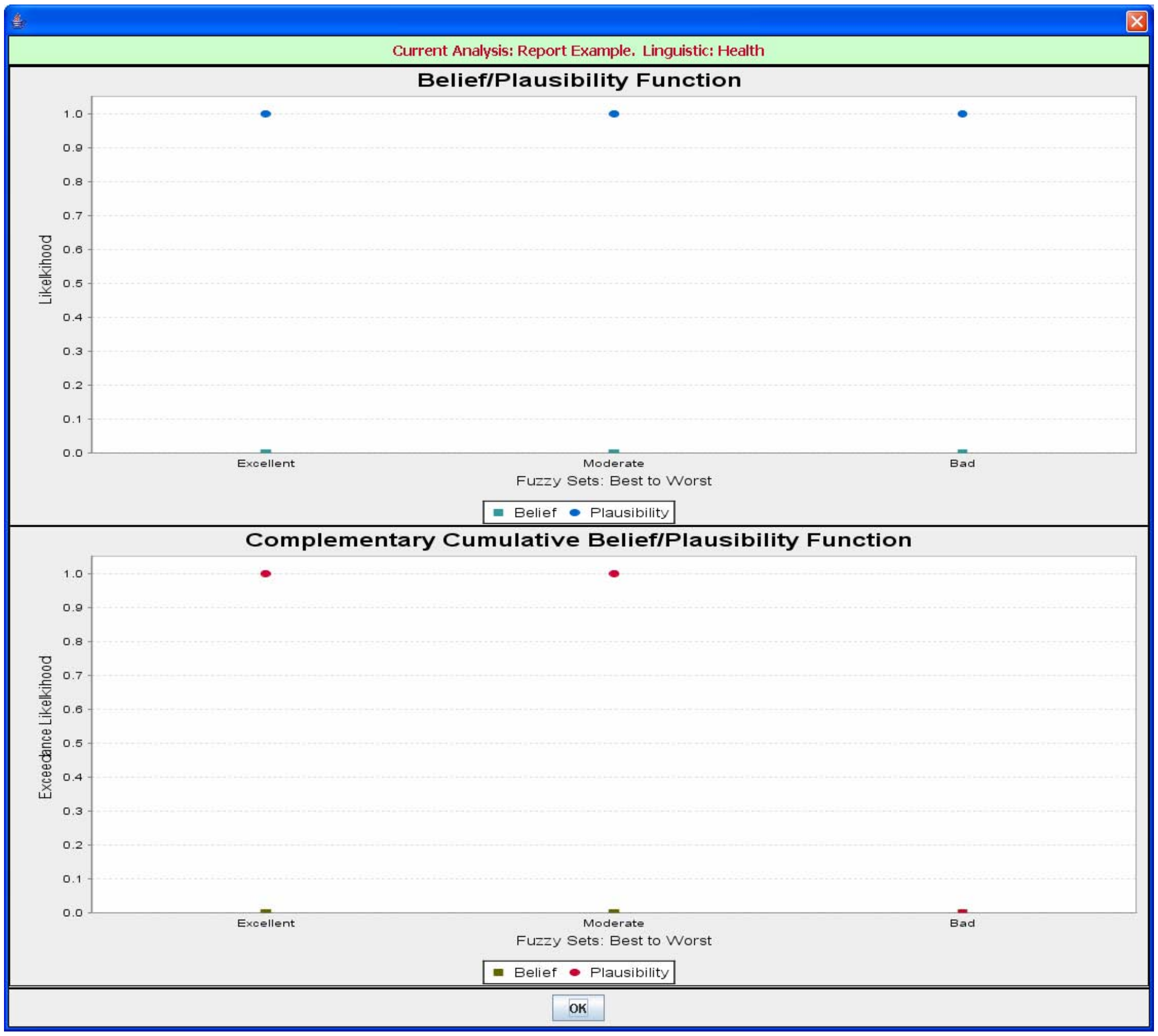


Figure 4-4. Graphical Results for "Health" for "Mary"

There is no (zero) belief for any of the fuzzy sets for the "Health" of "Mary," but it is totally plausible (one) that her "Health" can be "Bad," "Moderate," or "Excellent."

5 Summary

This report summarizes the use of the LinguisticBelief Java code for performing linguistic evaluations. Uncertainty is considered by segregating each linguistic variable into fuzzy sets. Linguistic variables are combined using an approximate reasoning rule base on the fuzzy sets. Uncertainty is captured and propagated using the belief/plausibility measure of uncertainty, thereby allowing considerable epistemic uncertainty to be considered. For situations where the uncertainty is purely aleatory, both belief and plausibility become the traditional probability measure of uncertainty.

Appendix A. Code Architecture and Deployment

Section A.1 summarizes the architecture of LinguisticBelief. Section A.2 summarizes the deployment package for LinguisticBelief.

A.1 Summary of Code Architecture

Figure A-1 shows the structure of the LinguisticBelief application in the NetBeans Integrated Development Environment (IDE). As indicated in the **Projects** panel in the left side of the figure, the application is segregated into numerous Java packages. The "datapackage" contains the classes used to perform calculations and the "guipackage" contains the classes used for gui. The "xmlJavatools" package contains the classes used for saving results in an xml (extended markup language) format.

The tool allows an analysis to be saved in both xml format and Java serialized (binary) format. The xml files are humanly readable; the binary files are not. The coding in LinguisticBelief required to implement handling data in xml is extensive; LinguisticBelief implements xml using a SAX parser. The coding required to implement serialization of data is simple, since serialization is built into Java. The "xmlpackage" contains all analyses saved in xml format. The "serializedAnalyses" package contains all analyses saved using serialization. Listing A-1 is a portion of the xml data file for the model discussed in Section 1.1.

The "Libraries" package indicates the libraries used by the application, including the JFreeChart library. Figure A-2 shows the JFreeChart library in the IDE.

The other packages shown in Figure A-1 are not pertinent to this discussion.

As indicated in Figure A-1, the Driver class contains the main method for the application.

A basic linguistic is implemented by the class datapackage.BasicLinguistic. A rule-based linguistic is implemented by the class datapackage.RuleLinguistic, which extends BasicLinguistic; thus, a RuleLinguistic "is-a" BasicLinguistic; that is, BasicLinguistic is a super class for the subclass RuleLinguistic, and RuleLinguistic inherits all the properties of BasicLinguistic.

Java is multithreaded. Since Java Swing is not thread safe, the Driver.main method runs the gui in the event dispatch thread, as indicated in Figure A-1.

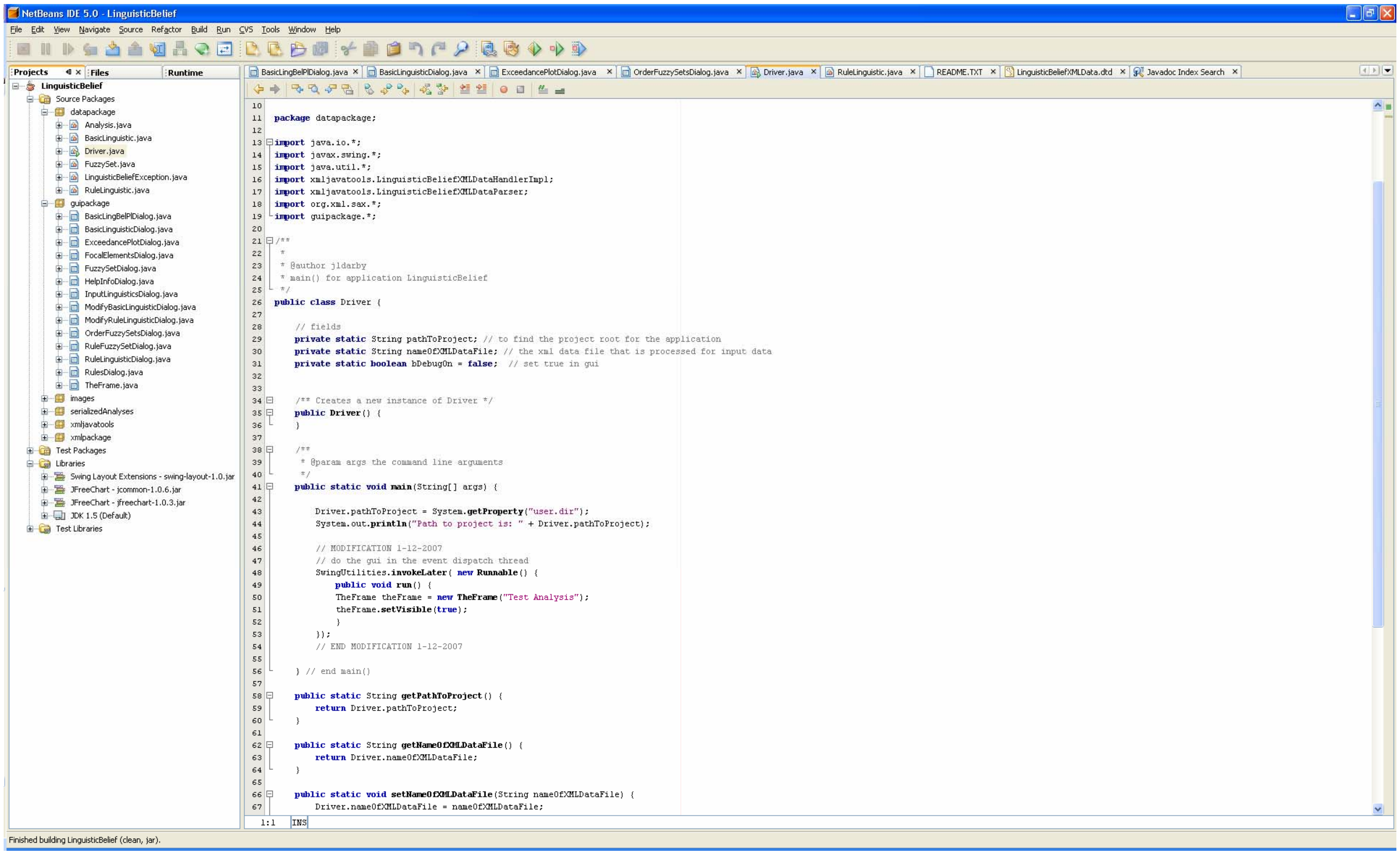
The class datapackage.LinguisticBeliefException extends the Java class Exception and is used to trap errors.

Numerous inner classes are used as appropriate; for example, Rule is an inner class declared within RuleLinguistic as shown in Figure A-3. Also, LinguisticBelief makes extensive use of generics (a feature new to Java as of JSDK5) for data structures. For example, as indicated in Figure A-3, RuleLinguistic has the data member

```
private ArrayList<BasicLinguistic> theInputLinguistics;
```

that declares the InputLinguistics as a reference to an ArrayList that contains instances of BasicLinguistic as elements. (ArrayList is a data type built into Java that allows dynamic creation and manipulation of a collection of elements.) LinguisticBelief will *not* run with versions of Java earlier than JSDK5.

The “guipackage” uses numerous standard Java Swing gui classes. The top-level gui element is a JFrame; for example, Figure 1-2 in the main report is a JFrame. The frame uses a JTree to display and allow the user to modify an analysis; for example, the left-side panel (a JPanel) in Figure 1-2 contains a JTree.


NetBeans IDE 5.0 - LinguisticBelief
File Edit View Navigate Source Refactor Build Run CVS Tools Window Help
Projects
Files
Runtime
BasicLingBelPlDialog.java
BasicLinguisticDialog.java
ExceedancePlotDialog.java
OrderFuzzySetsDialog.java
Driver.java
RuleLinguistic.java
README.TXT
LinguisticBeliefXMLData.dtd
Javadoc Index Search
1:1
INS

```
LinguisticBelief
  Source Packages
    datapackage
      Analysis.java
      BasicLinguistic.java
      Driver.java
      FuzzySet.java
      LinguisticBeliefException.java
      RuleLinguistic.java
    guipackage
      BasicLingBelPlDialog.java
      BasicLinguisticDialog.java
      ExceedancePlotDialog.java
      FocalElementsDialog.java
      FuzzySetDialog.java
      HelpInfoDialog.java
      InputLinguisticsDialog.java
      ModifyBasicLinguisticDialog.java
      ModifyRuleLinguisticDialog.java
      OrderFuzzySetsDialog.java
      RuleFuzzySetDialog.java
      RuleLinguisticDialog.java
      RulesDialog.java
      TheFrame.java
    images
    serializedAnalyses
    xmljavatools
    xmlpackage
  Test Packages
  Libraries
    Swing Layout Extensions - swing-layout-1.0.jar
    JFreeChart - jcommon-1.0.6.jar
    JFreeChart - jfreechart-1.0.3.jar
    JDK 1.5 (Default)
  Test Libraries
```

```java

package datapackage;

import java.io.*;
import javax.swing.*;
import java.util.*;
import xmljavatools.LinguisticBeliefXMLDataHandlerImpl;
import xmljavatools.LinguisticBeliefXMLDataParser;
import org.xml.sax.*;
import guipackage.*;

/**
 *
 * @author jldarby
 * main() for application LinguisticBelief
 */
public class Driver {

    // fields
    private static String pathToProject; // to find the project root for the application
    private static String nameOfXMLDataFile; // the xml data file that is processed for input data
    private static boolean bDebugOn = false;  // set true in gui


    /** Creates a new instance of Driver */
    public Driver() {
    }

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {

        Driver.pathToProject = System.getProperty("user.dir");
        System.out.println("Path to project is: " + Driver.pathToProject);

        // MODIFICATION 1-12-2007
        // do the gui in the event dispatch thread
        SwingUtilities.invokeLater( new Runnable() {
            public void run() {
            TheFrame theFrame = new TheFrame("Test Analysis");
            theFrame.setVisible(true);
            }
        });
        // END MODIFICATION 1-12-2007

    } // end main()

    public static String getPathToProject() {
        return Driver.pathToProject;
    }

    public static String getNameOfXMLDataFile() {
        return Driver.nameOfXMLDataFile;
    }

    public static void setNameOfXMLDataFile(String nameOfXMLDataFile) {
        Driver.nameOfXMLDataFile = nameOfXMLDataFile;
```

Finished building LinguisticBelief (clean, jar).

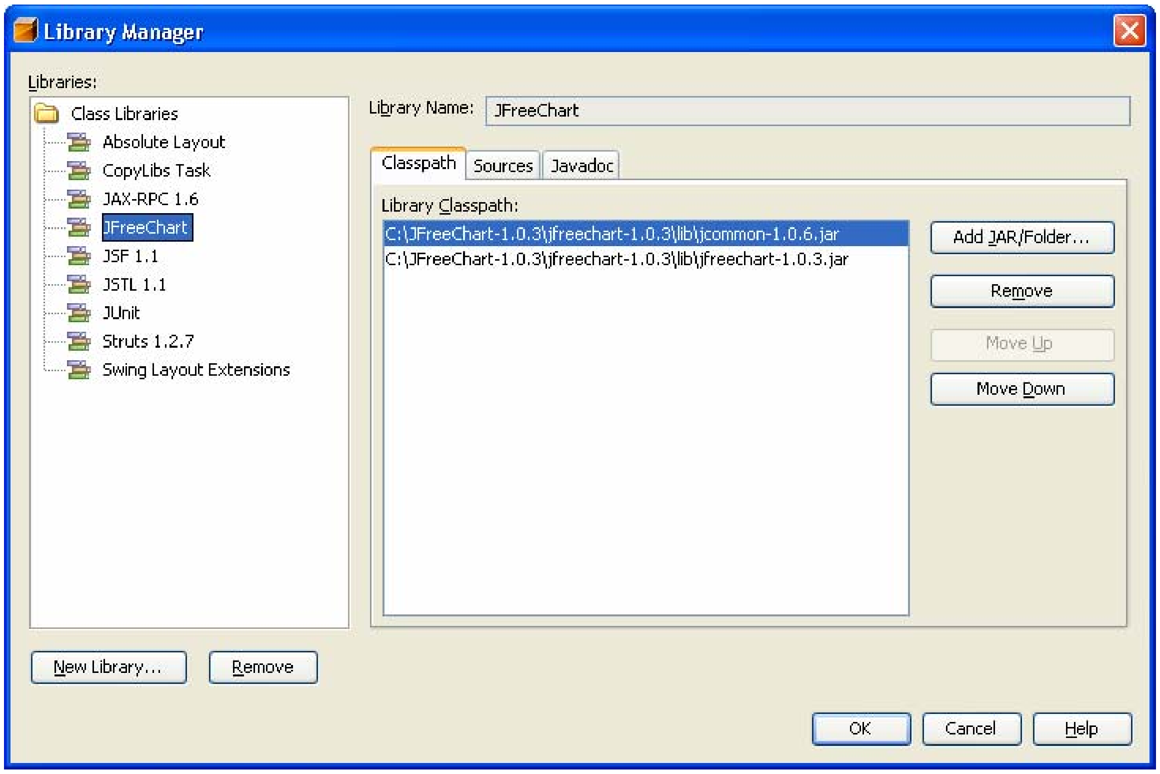
Figure A-1. LinguisticBelief in the NetBeans IDE

Listing A-1. Portion of an Analysis Saved in xml Format

```xml
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE linguisticBeliefData SYSTEM "LinguisticBeliefXMLData.dtd" >
<!-- this xml file was generated by Analysis.dumpAnalysisToXMLFile() -->
<linguisticBeliefData>
 <!-- Name Of Analysis -->
<analysisName>Report Example</analysisName>
 <!-- all the BasicLinguistics used in Analysis -->
<basicLinguistic>
<basicLinguisticName>Health</basicLinguisticName>
<basicLinguisticFuzzySet>Bad</basicLinguisticFuzzySet>
<basicLinguisticFuzzySet>Moderate</basicLinguisticFuzzySet>
<basicLinguisticFuzzySet>Excellent</basicLinguisticFuzzySet>
<basicLinguisticFocalElement>
<focalElementFuzzySet>Bad</focalElementFuzzySet>
<focalElementFuzzySet>Moderate</focalElementFuzzySet>
<evidence>0.800</evidence></basicLinguisticFocalElement>
<basicLinguisticFocalElement>
<focalElementFuzzySet>Moderate</focalElementFuzzySet>
<focalElementFuzzySet>Excellent</focalElementFuzzySet>
<evidence>0.200</evidence></basicLinguisticFocalElement>
</basicLinguistic>
<basicLinguistic>
<basicLinguisticName>Wealth</basicLinguisticName>
<basicLinguisticFuzzySet>Poor</basicLinguisticFuzzySet>
<basicLinguisticFuzzySet>Middle Class</basicLinguisticFuzzySet>
<basicLinguisticFuzzySet>Rich</basicLinguisticFuzzySet>
<basicLinguisticFocalElement>
<focalElementFuzzySet>Middle Class</focalElementFuzzySet>
<evidence>0.300</evidence></basicLinguisticFocalElement>
<basicLinguisticFocalElement>
<focalElementFuzzySet>Poor</focalElementFuzzySet>
<focalElementFuzzySet>Middle Class</focalElementFuzzySet>
<evidence>0.700</evidence></basicLinguisticFocalElement>
</basicLinguistic>
<basicLinguistic>
<basicLinguisticName>Outlook on Life</basicLinguisticName>
<basicLinguisticFuzzySet>Pessimist</basicLinguisticFuzzySet>
<basicLinguisticFuzzySet>Optimist</basicLinguisticFuzzySet>
<basicLinguisticFocalElement>
<focalElementFuzzySet>Pessimist</focalElementFuzzySet>
<evidence>0.020</evidence></basicLinguisticFocalElement>
<basicLinguisticFocalElement>
<focalElementFuzzySet>Pessimist</focalElementFuzzySet>
<focalElementFuzzySet>Optimist</focalElementFuzzySet>
<evidence>0.980</evidence></basicLinguisticFocalElement>
</basicLinguistic>
```

```
<!-- all the RuleLinguistics used in Analysis -->
<!-- must be ordered such that current RuleLinguistic does not depend on a later RuleLinguistic -->
<ruleLinguistic>
<ruleLinguisticName>Quality of Life</ruleLinguisticName>
<ruleLinguisticFuzzySet>Good</ruleLinguisticFuzzySet>
<ruleLinguisticFuzzySet>Not so Good</ruleLinguisticFuzzySet>
<ruleLinguisticBasicLinguisticInput>Health</ruleLinguisticBasicLinguisticInput>
<ruleLinguisticBasicLinguisticInput>Wealth</ruleLinguisticBasicLinguisticInput>
<rule>
<ruleInput>
<ruleInputBasicLinguisticName>Health</ruleInputBasicLinguisticName>
<ruleInputBasicLinguisticFuzzySet>Bad</ruleInputBasicLinguisticFuzzySet>
</ruleInput>
<ruleInput>
<ruleInputBasicLinguisticName>Wealth</ruleInputBasicLinguisticName>
<ruleInputBasicLinguisticFuzzySet>Poor</ruleInputBasicLinguisticFuzzySet>
</ruleInput>
<ruleOutputFuzzySet>Not so Good</ruleOutputFuzzySet>
</rule>
```

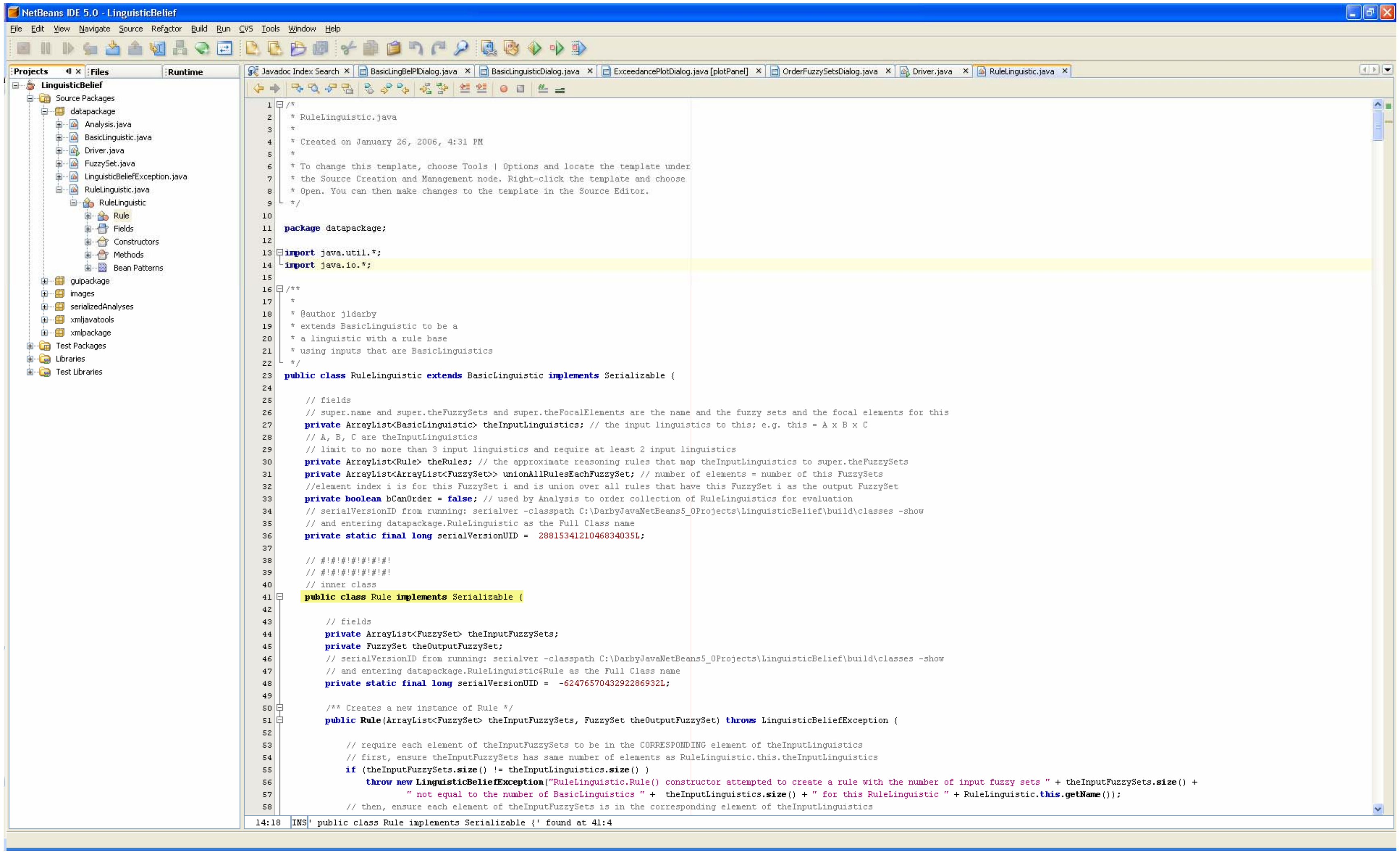
(Rest of file not shown here)



Library Name: JFreeChart

```
C:\JFreeChart-1.0.3\jfreechart-1.0.3\lib\jcommon-1.0.6.jar
C:\JFreeChart-1.0.3\jfreechart-1.0.3\lib\jfreechart-1.0.3.jar
```

Figure A-2. JFreeChart as a Library in the IDE



```
/*
 * RuleLinguistic.java
 *
 * Created on January 26, 2006, 4:31 PM
 *
 * To change this template, choose Tools | Options and locate the template under
 * the Source Creation and Management node. Right-click the template and choose
 * Open. You can then make changes to the template in the Source Editor.
 */

package datapackage;

import java.util.*;
import java.io.*;

/**
 *
 * @author jldarby
 * extends BasicLinguistic to be a
 * a linguistic with a rule base
 * using inputs that are BasicLinguistics
 */
public class RuleLinguistic extends BasicLinguistic implements Serializable {

    // fields
    // super.name and super.theFuzzySets and super.theFocalElements are the name and the fuzzy sets and the focal elements for this
    private ArrayList<BasicLinguistic> theInputLinguistics; // the input linguistics to this; e.g. this = A x B x C
    // A, B, C are theInputLinguistics
    // limit to no more than 3 input linguistics and require at least 2 input linguistics
    private ArrayList<Rule> theRules; // the approximate reasoning rules that map theInputLinguistics to super.theFuzzySets
    private ArrayList<ArrayList<FuzzySet>> unionAllRulesEachFuzzySet; // number of elements = number of this FuzzySets
    //element index i is for this FuzzySet i and is union over all rules that have this FuzzySet i as the output FuzzySet
    private boolean bCanOrder = false; // used by Analysis to order collection of RuleLinguistics for evaluation
    // serialVersionID from running: serialver -classpath C:\DarbyJavaNetBeans5_0Projects\LinguisticBelief\build\classes -show
    // and entering datapackage.RuleLinguistic as the Full Class name
    private static final long serialVersionUID =  2881534121046834035L;

    // #!#!#!#!#!#!#!
    // #!#!#!#!#!#!#!
    // inner class
    public class Rule implements Serializable {

        // fields
        private ArrayList<FuzzySet> theInputFuzzySets;
        private FuzzySet theOutputFuzzySet;
        // serialVersionID from running: serialver -classpath C:\DarbyJavaNetBeans5_0Projects\LinguisticBelief\build\classes -show
        // and entering datapackage.RuleLinguistic$Rule as the Full Class name
        private static final long serialVersionUID =  -6247657043292286932L;

        /** Creates a new instance of Rule */
        public Rule(ArrayList<FuzzySet> theInputFuzzySets, FuzzySet theOutputFuzzySet) throws LinguisticBeliefException {

            // require each element of theInputFuzzySets to be in the CORRESPONDING element of theInputLinguistics
            // first, ensure theInputFuzzySets has same number of elements as RuleLinguistic.this.theInputLinguistics
            if (theInputFuzzySets.size() != theInputLinguistics.size() )
                throw new LinguisticBeliefException("RuleLinguistic.Rule() constructor attempted to create a rule with the number of input fuzzy sets " + theInputFuzzySets.size() +
                        " not equal to the number of BasicLinguistics " +  theInputLinguistics.size() + " for this RuleLinguistic " + RuleLinguistic.this.getName());
            // then, ensure each element of theInputFuzzySets is in the corresponding element of theInputLinguistics
```

```
14:18  INS ' public class Rule implements Serializable {' found at 41:4
```

Figure A-3. Rule as an Inner Class in RuleLinguistic

LinguisticBelief uses recursion to effect changes to an analysis. For example, the gui class datapackage.RuleLinguistic uses the RuleLinguistic.deleteAllRulesAndInputLinguisticIfInputLinguistic() method in Listing A-2 to respond to a user request to delete a previously created RuleLinguistic.

Listing A-2. Recursive Method for Removing a RuleLinguistic

```
 /** recursively delete rules/focal elements if this has theInputLinguistic as an input linguistic
 * (but do not delete theInputLinguistic as an input linguistic for this since that is done in the gui calling method)
 */
 public void deleteAllRulesAndInputLinguisticIfInputLinguistic(RuleLinguistic topRuleLinguistic, RuleLinguistic currentRuleLinguistic,
BasicLinguistic theInputLinguisticToDelete) {
    // topRuleLinguistic is the RuleLinguistic of interest and is not changed in this method
    // currentRuleLinguistic is any RuleLinguistic that is an input into topRuleLinguistic
    // theInputLinguisticToDelete is the input linguistic to be deleted and is not changed in this method
    if ( Driver.isDebugOn() ) {
        System.out.println("    deleteAllRulesAndInputLinguisticIfInputLinguistic() looking at: " + currentRuleLinguistic);
    }
    Iterator<BasicLinguistic> iterInputLing = currentRuleLinguistic.getTheInputLinguistics().iterator();
    while (iterInputLing.hasNext() ) {
        BasicLinguistic anInputLinguistic = iterInputLing.next();

        if ( anInputLinguistic.getClass() == RuleLinguistic.class ) {  // this may have an input with an input that is theInputLinguistic
            if ( ( (RuleLinguistic) anInputLinguistic).getTheFocalElements().isEmpty() ) { // anInputLinguistic may have had focal elements removed
            //  from operation on another currentRuleLinguistic
                if ( Driver.isDebugOn() ) {
                    System.out.println("       deleteAllRulesAndInputLinguisticIfInputLinguistic() has a RuleLinguistic: " + anInputLinguistic +
                    " with no focal elements.  So, remove rules and focal elements from: " + topRuleLinguistic);
                }
                topRuleLinguistic.theRules = new ArrayList<Rule> ();
                topRuleLinguistic.emptyFocalElements();
                topRuleLinguistic.emptyBeliefPlausibility();
            }
            if ( Driver.isDebugOn() ) {
                System.out.println("       deleteAllRulesAndInputLinguisticIfInputLinguistic() operating recursively on: " + anInputLinguistic);
            }
            ( (RuleLinguistic) anInputLinguistic ).deleteAllRulesAndInputLinguisticIfInputLinguistic(topRuleLinguistic, (RuleLinguistic)
            anInputLinguistic, theInputLinguisticToDelete); // recursive call
        }

        if (anInputLinguistic.equals(theInputLinguisticToDelete)) { // theInputLinguistic itself is an input for this
            // remove all rules
            topRuleLinguistic.theRules = new ArrayList<Rule> ();
            if ( Driver.isDebugOn() ) {
```

```
            System.out.println("            deleteAllRulesAndInputLinguisticIfInputLinguistic() removed focal elements from " + topRuleLinguistic);
          }
          // remove rules from topRuleLinguistic
          topRuleLinguistic.emptyFocalElements();
          topRuleLinguistic.emptyBeliefPlausibility();
          if ( Driver.isDebugOn() ) {
            System.out.println("            deleteAllRulesAndInputLinguisticIfInputLinguistic() removed rules from " + topRuleLinguistic);
          }

          return;
        } // if
      } // end while iterInputLing
```

As discussed in a reference, the code performs an automatic "aggregation" of focal elements as it progresses through the evaluation of rule-based linguistic variables. [Darby]

This aggregation is best discussed by an example. Consider the rule base for the linguistics "Quality Of Life" and "Happiness" presented in Section 1.1. Using the LinguisticBelief Code the following results are obtained (selected output presented) with Utilities|DebugOn selected.

```
Quality of Life Focal Elements

For RuleLinguistic Quality of Life a FocalElement with degree of evidence 2.4000e-01 is:
   Bad_&_Middle Class
   Moderate_&_Middle Class

For RuleLinguistic Quality of Life a FocalElement with degree of evidence 5.6000e-01 is:
   Bad_&_Poor
   Bad_&_Middle Class
   Moderate_&_Poor
   Moderate_&_Middle Class

For RuleLinguistic Quality of Life a FocalElement with degree of evidence 6.0000e-02 is:
   Moderate_&_Middle Class
   Excellent_&_Middle Class

For RuleLinguistic Quality of Life a FocalElement with degree of evidence 1.4000e-01 is:
   Moderate_&_Poor
   Moderate_&_Middle Class
   Excellent_&_Poor
   Excellent_&_Middle Class

Quality of Life Rule Base

For RuleLinguistic Quality of Life the fuzzy sets from union of rules with output FuzzySet Not So Good
are:
   Bad_&_Poor
   Moderate_&_Poor
   Bad_&_Middle Class
   Moderate_&_Middle Class
   Bad_&_Rich

For RuleLinguistic Quality of Life the fuzzy sets from union of rules with output FuzzySet Good are:
   Excellent_&_Poor
   Excellent_&_Middle Class
   Moderate_&_Rich
   Excellent_&_Rich

Quality of Life Belief / Plausibility Intervals

For BasicLinguistic Quality of Life For fuzzy set Not So Good Belief / Plausibility interval is: 8.00000e-01 /
1.00000e+00
For BasicLinguistic Quality of Life For fuzzy set Good Belief / Plausibility interval is: 0.00000e+00 /
2.00000e-01
```

```
Outlook on Life Belief / Plausibility Intervals

For BasicLinguistic Outlook On Life For fuzzy set Pessimist Belief / Plausibility interval is 2.0000e-02 ,
1.0000e+00
For BasicLinguistic Outlook On Life For fuzzy set Optimist Belief / Plausibility interval is 0.0000e+00 ,
9.8000e-01
```

To evaluate "Happiness" using the rule base, its focal elements need to be expressed in terms of the fuzzy sets of its input variables, which is accomplished by aggregation. For example, before aggregation the focal elements for "Happiness" are:

```
(1) For RuleLinguistic Happiness a FocalElement with degree of evidence: 4.8000e-03 is
    Bad_&_Middle Class_&_Pessimist
    Moderate_&_Middle Class_&_Pessimist

(2) For RuleLinguistic Happiness a FocalElement with degree of evidence: 2.3520e-01 is
    Bad_&_Middle Class_&_Pessimist
    Bad_&_Middle Class_&_Optimist
    Moderate_&_Middle Class_&_Pessimist
    Moderate_&_Middle Class_&_Optimist

(3) For RuleLinguistic Happiness a FocalElement with degree of evidence 1.1200e-02 is:
    Bad_&_Poor_&_Pessimist
    Bad_&_Middle Class_&_Pessimist
    Moderate_&_Poor_&_Pessimist
    Moderate_&_Middle Class_&_Pessimist


(4) For RuleLinguistic Happiness a FocalElement with degree of evidence 5.4880e-01 is:
    Bad_&_Poor_&_Pessimist
    Bad_&_Poor_&_Optimist
    Bad_&_Middle Class_&_Pessimist
    Bad_&_Middle Class_&_Optimist
    Moderate_&_Poor_&_Pessimist
    Moderate_&_Poor_&_Optimist
    Moderate_&_Middle Class_&_Pessimist
    Moderate_&_Middle Class_&_Optimist

(5) For RuleLinguistic Happiness a FocalElement with degree of evidence 1.2000e-03 is:
    Moderate_&_Middle Class_&_Pessimist
    Excellent_&_Middle Class_&_Pessimist

(6) For RuleLinguistic Happiness a FocalElement with degree of evidence 5.8800e-02 is:
    Moderate_&_Middle Class_&_Pessimist
    Moderate_&_Middle Class_&_Optimist
    Excellent_&_Middle Class_&_Pessimist
    Excellent_&_Middle Class_&_Optimist

(7) For RuleLinguistic Happiness a FocalElement with degree of evidence 2.8000e-03 is:
    Moderate_&_Poor_&_Pessimist
    Moderate_&_Middle Class_&_Pessimist
    Excellent_&_Poor_&_Pessimist
    Excellent_&_Middle Class_&_Pessimist

(8) For RuleLinguistic Happiness a FocalElement with degree of evidence 1.3720e-01 is:
    Moderate_&_Poor_&_Pessimist
    Moderate_&_Poor_&_Optimist
    Moderate_&_Middle Class_&_Pessimist
    Moderate_&_Middle Class_&_Optimist
    Excellent_&_Poor_&_Pessimist
    Excellent_&_Poor_&_Optimist
    Excellent_&_Middle Class_&_Pessimist
    Excellent_&_Middle Class_&_Optimist
```

The rule base for “Happiness” is:

```
For RuleLinguistic Happiness the fuzzy sets from union of rules with output FuzzySet Very Happy are:
   Good_&_Optimist

For RuleLinguistic Happiness the fuzzy sets from union of rules with output FuzzySet Accepting are:
   Not So Good_&_Optimist
   Good_&_Pessimist

For RuleLinguistic Happiness the fuzzy sets from union of rules with output FuzzySet Depressed are:
   Not So Good_&_Pessimist
```

The focal elements for “Happiness” contain “Quality Of Life” in terms of the fuzzy sets of its input variables “Health” and “Wealth,” but the rule base for “Happiness” is expressed in terms of the fuzzy sets for “Quality Of Life.” The focal elements for “Happiness” must be aggregated to be expressed in terms of the fuzzy sets of “Quality Of Life” instead of being expressed in terms of the fuzzy sets for the input variables for “Quality Of Life.”

Aggregation is performed in two steps. First, for any given focal element, its constituent input fuzzy sets are mapped to the appropriate output fuzzy set per the rule base. For “Happiness,” this results in the following focal elements:

```
(1) For RuleLinguistic Happiness a FocalElement with degree of evidence 4.8000e-03 is:
   Not So Good_&_Pessimist

(2) For RuleLinguistic Happiness a FocalElement with degree of evidence 2.3520e-01 is:
   Not So Good_&_Pessimist
   Not So Good_&_Optimist

(3) For RuleLinguistic Happiness a FocalElement with degree of evidence 1.1200e-02 is:
   Not So Good_&_Pessimist

(4) For RuleLinguistic Happiness a FocalElement with degree of evidence: 5.4880e-01 is
   Not So Good_&_Pessimist
   Not So Good_&_Optimist

(5) For RuleLinguistic Happiness a FocalElement with degree of evidence 1.2000e-03 is:
   Not So Good_&_Pessimist
   Good_&_Pessimist

(6) For RuleLinguistic Happiness a FocalElement with degree of evidence 5.8800e-02 is:
   Not So Good_&_Pessimist
   Not So Good_&_Optimist
   Good_&_Pessimist
   Good_&_Optimist

(7) For RuleLinguistic Happiness a FocalElement with degree of evidence 2.8000e-03 is:
   Not So Good_&_Pessimist
   Good_&_Pessimist

(8) For RuleLinguistic Happiness a FocalElement with degree of evidence 1.3720e-01 is:
   Not So Good_&_Pessimist
   Not So Good_&_Optimist
   Good_&_Pessimist
   Good_&_Optimist
```

Second, focal elements resulting from step one containing identical fuzzy sets are combined by adding the degrees of evidence, resulting in the following focal elements for "Happiness."

```
(A) For RuleLinguistic Happiness a FocalElement with degree of evidence 1.6000e-02 is:
   Not So Good_&_Pessimist

(B) For RuleLinguistic Happiness a FocalElement with degree of evidence 7.8400e-01 is:
   Not So Good_&_Pessimist
   Not So Good_&_Optimist

(C) For RuleLinguistic Happiness a FocalElement with degree of evidence 4.0000e-03 is:
   Not So Good_&_Pessimist
   Good_&_Pessimist

(D) For RuleLinguistic Happiness a FocalElement with degree of evidence 1.9600e-01 is:
   Not So Good_&_Pessimist
   Not So Good_&_Optimist
   Good_&_Pessimist
   Good_&_Optimist
```

This two-step aggregation operation reduced the number of focal elements for "Happiness" from eight to four.

The most complex algorithm in the code is associated with calculating the focal elements for a Rule Linguistic from the focal elements of its constituent input linguistics. This algorithm uses two nested recursive methods to allow any number of input linguistics to be evaluated. Listing A-3 is the RuleLinguistic.calculateFocalElements() method that implements that algorithm.

Listing A-3. Method with Algorithm for Evaluating Focal Elements for a Rule Linguistic

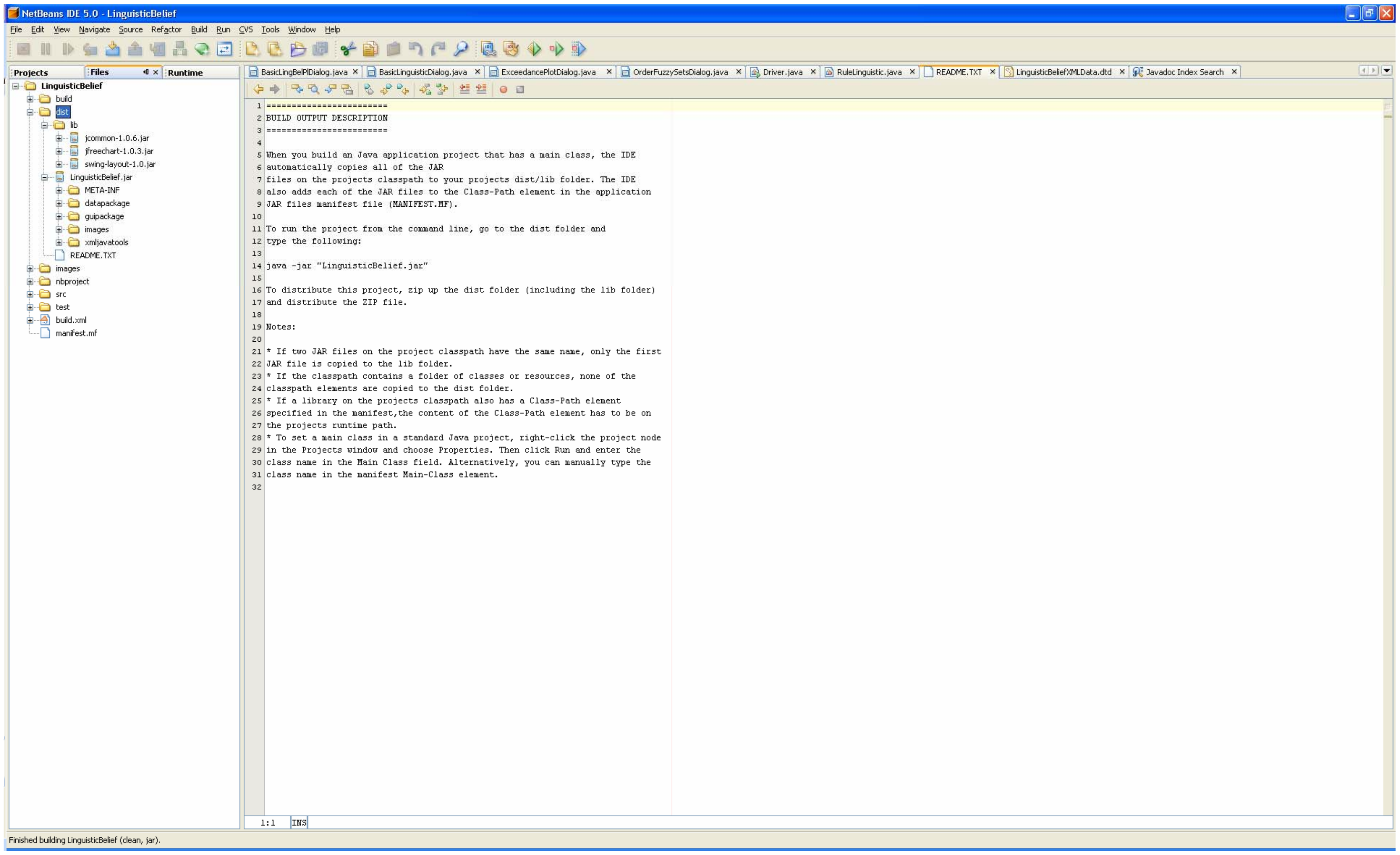
```
/** RuleLinguistic.calculateFocalElements(): calculate focal elements assuming noninteraction */
  public void calculateFocalElements() {

      // Consolidate focal elements for each input that is a RuleLinguistic.
      // For each input that is a BasicLinguistic consolidate does nothing
      Iterator<BasicLinguistic> iterInputLinguistics = this.theInputLinguistics.iterator();
      while (iterInputLinguistics.hasNext()) {
         BasicLinguistic inputLinguistic = iterInputLinguistics.next();
         inputLinguistic.consolidateFocalElements();
      }

      // form this.theFocalElements from each this.theInputLinguistics focal elements

     // allow any number of input linguistics using recursion

      // create a collection to hold a collection of FocalElements for each input linguistic
      ArrayList<ArrayList<FocalElement>> collectionInputFocalElements = new
ArrayList<ArrayList<FocalElement>> ();
      // fill collectionInputFocalElements
      Iterator<BasicLinguistic> iterInputs = this.theInputLinguistics.iterator();
      while (iterInputs.hasNext()) {
         collectionInputFocalElements.add(iterInputs.next().getTheFocalElements());
      } // end while
      if ( Driver.isDebugOn() ) {
         System.out.println("");
          System.out.println("the RuleLinguistic " + this + " called RiskLinguistic.calculateFocalElements()
and set collectionInputFocalElements to
          be: " + collectionInputFocalElements);
      }
      // EXAMPLE this.RuleLinguistic, call it D, has 3 basicLinguistic inputs A, B, anc C
      // A has focal elements A1 & A2, A1 with evidence 0.2, 0.8
      // B has focal elements B1 & B2, B2 with evidence 0.8, 0.2
      // C has focal elements C1, C2 with evidence 0.5, 0.5
      // collectionInputFocalElements for D is: [[A1, A1 & A2], [B2, B1 & B2], [C1, C2]]

      // operate on collectionInputFocalElements to calculate the focal elements for this
      // the result for EXAMPLE D will have 8 focal elements with the following fuzzy sets
      /*
      A1_&_B2_&_C1 with evidence 8.0000e-02
      A1_&_B2_&_C2 with evidence 8.0000e-02
      A1_&_B1_&_C1 & A1_&_B2_&_C1 with evidence 3.2000e-01
      A1_&_B1_&_C2 & A1_&_B2_&_C2 with evidence 3.2000e-01
      A1_&_B2_&_C1 & A2_&_B2_&_C1 with evidence 8.0000e-02
      A1_&_B2_&_C2 & A2_&_B2_&_C2 with evidence 8.0000e-02
      A1_&_B1_&_C1 & A1_&_B2_&_C1 & A2_&_B1_&_C1 & A2_&_B2_&_C1 with evidence 8.0000e-02
      A1_&_B1_&_C2 & A1_&_B2_&_C2 & A2_&_B1_&_C2 & A2_&_B2_&_C2 with evidence 8.0000e-02
      where '_&_' denotes combinations of fuzzy sets from each ordered input linguistic
      and '&' denotes combinations for the resulting this.RuleLinguistic
       */

       if ( Driver.isDebugOn() ) {
```

```
                System.out.println("    RiskLinguistic.calculateFocalElements() is CALLING
        formFocalElements() to operate on
                collectionInputFocalElements");
      }

      formFocalElements(collectionInputFocalElements, 0); // formFocalElements() is recursive, and
internally formFocalElements()
      // calls setFuzzySetsForOneFocalElement() which is also recursive

      if ( Driver.isDebugOn() ) {
         System.out.println("    formFocalElements() RETURNED");
         System.out.println("RiskLinguistic.calculateFocalElements() RETURNED");
      }

      // set this RuleLinguistic focal elements
      this.setTheFocalElements(this.theFocalElementsForThisRuleLinguistic);

      if ( Driver.isDebugOn() ) {
         System.out.println();
         System.out.println("RuleLinguistic " + RuleLinguistic.this.getName() + " has " +
this.getTheFocalElements().size() +
                " focal elements as follows");
         Iterator<FocalElement> iterFocalElements = theFocalElementsForThisRuleLinguistic.iterator();
         while (iterFocalElements.hasNext()) {
            FocalElement aFocalElement = iterFocalElements.next();
            System.out.println();
            System.out.printf(" For RuleLinguistic " + RuleLinguistic.this.getName() + " a FocalElement with
degree of evidence %.4e is: \n,"
                aFocalElement.getTheEvidence());
            Iterator<FuzzySet> iterFuzzySets = aFocalElement.getTheFuzzySets().iterator();
            while (iterFuzzySets.hasNext()) {
               FuzzySet aFuzzySet = iterFuzzySets.next();
               System.out.println("    " + aFuzzySet.getName());
            } // end inner while
         } // end outer while
      }

   } // end calculateFocalElements()

   /* RuleLinguistic.formFocalElements() */
   // used by this.calculateFocalElements() to allow any number of input linguistics for this.
   // this.calculateFocalelements() passes in collectionInputFocalElements, a collection of
   // focal elements for each input in order; e.g., index 0 element is focal elements for input 1, etc.
   // Recursively operate on all input focal elements
   // first call is with nIndexInput = 0
   private void formFocalElements(ArrayList<ArrayList<FocalElement>> collectionInputFocalElements, int
nIndexInput) {
      // EXAMPLE collectioncollectionInputFocalElements for D is: [[A1, A1 & A2], [B2, B1 & B2], [C1, C2]]

      // if index is one more than number of inputs
      // so we are done since we have completed evaluation all collectionInputFocalElements
      if (nIndexInput == collectionInputFocalElements.size() ) {
         return;
      }

      if (nIndexInput == 0) {
            // initialize variables to be calculated
            this.theFocalElementsForThisRuleLinguistic = new ArrayList<FocalElement> ();
            // FocalElements for this formed from FocalElements for inputs to this
        }

        // get the focal elements for input at nIndexInput
        Iterator<FocalElement> iterAnInputFocalElements =
collectionInputFocalElements.get(nIndexInput).iterator();
        while ( iterAnInputFocalElements.hasNext() ) {

            // this.formulatingFocalElement.size() is a private data member of RuleLinguistic initially set to
new per its declaration
            // remove all elements from end back through nIndexInput from this.formulatingFocalElement to
reset for next iteration
            for (int nRemove = this.formulatingFocalElement.size() - 1; nRemove >= nIndexInput; nRemove--)
{

                if ( Driver.isDebugOn() ) {
                    System.out.println("       RESETTING this.formulatingFocalElement.size(): " +
this.formulatingFocalElement.size() +
                        "   for nIndexInput: " + nIndexInput + " .... Removing input index: " + nRemove);
                }

                this.formulatingFocalElement.remove(nRemove);
            }

            FocalElement anInputFocalElement = iterAnInputFocalElements.next();
            // add to this.formulatingFocalelement
            this.formulatingFocalElement.add(anInputFocalElement);

            // #!#!#!#!##!#!#!#
            // create new focal element for this if nIndexInput is last element of collection
            // #!#!#!#!##!#!#!#
            if (nIndexInput == collectionInputFocalElements.size() - 1) {
                // ASSERT this.formulatingFocalElement has size = collectionInputFocalElements.size()
                // and this.formulatingFocalElement contains focal elements from inputs with
                // index 'n-1' a focal element from input 'n'

                // form a focal element for this RuleLinguistic from the focal elements of the 'n' inputs
                if ( Driver.isDebugOn() ) {
                        System.out.println("       formFocalElements() is CALLING
             setFuzzySetsForOneFocalElement() with
                        this.formulatingFocalElement: " + this.formulatingFocalElement);
                }
                this.setFuzzySetsForOneFocalElement(this.formulatingFocalElement, 0); // recursive
                if ( Driver.isDebugOn() ) {
                    System.out.println("       setFuzzySetsForOneFocalElement() RETURNED");
                }

                try {
                    this.theFocalElementsForThisRuleLinguistic.add(new
FocalElement(this.aFocalElementForThisRuleLinguistic,
                        this.calculatedDegreeOfEvidence));
                    if ( Driver.isDebugOn() ) {
                        System.out.println("       this.RuleLinguistic currently has the following focal element:");
                        for (FocalElement focElem: this.theFocalElementsForThisRuleLinguistic)
                        System.out.println("           " + focElem);
                    }
                } catch (LinguisticBeliefException lbex) {
                }

            } // end if (nIndexInput == collectionInputFocalElements.size() - 1)
            // #!#!#!#!##!#!#!#
            // END create new focal element for this if nIndexInput is last element of collection
            // #!#!#!#!##!#!#!#

            // recursively continue with the focal elements at nIndexInput
            formFocalElements(collectionInputFocalElements, nIndexInput + 1 );

        } /// end while iterAnInputFocalElements

    } // end formFocalElements

    /* RuleLinguistic.setFuzzySetsForOneFocalElement() */
    // used by this.formFocalElements() to operate in this.formulatingFocalElement
    // Recursively operate on all elements of this.formulatingFocalElement
    // first call is with nIndexInput = 0.
    // collectionOneFocalElementFromInputs is a collection of focal elements that will form one
    // focal element for this.  collectionOneFocalElementFromInputs has one focal element from each input
to this,
// ordered; e.g. index 0 is a focal element from input 1.
// setFuzzySetsForOneFocalElement is called with formulatingFocalElement as arg for
collectionOneFocalElementFromInputs.
// degreeOfEvidence is degree of evidence ffo the new fuzzy set for this previously calculated.
private void setFuzzySetsForOneFocalElement(ArrayList<FocalElement> formulatingFocalElement, int
nIndexInput) {

        // if index is one more than number of inputs
        // so we are done since we have completed evaluation all formulatingFocalElement
        if (nIndexInput == formulatingFocalElement.size() ) {
            return;
        }

        // if index is 0 this is first call so start a new collection of fuzzy sets that will be filled and added
        // as a focal element to this
        // als calculate the degreeOfEvidence for the focal element for thie
        if (nIndexInput == 0) {
            this.aFocalElementForThisRuleLinguistic = new ArrayList<FuzzySet> ();
            // calculate the degreeOfEvidence for the new focal element
            this.calculatedDegreeOfEvidence = 1.0; // degree of evidence for new focal element for this
            Iterator<FocalElement> iterFocalElem = formulatingFocalElement.iterator();
            while (iterFocalElem.hasNext() ) {
                this.calculatedDegreeOfEvidence *= iterFocalElem.next().getTheEvidence();
            } // while iterFocalElem
        }

        // get a fuzzy set from the focal element at nIndexInput
        Iterator<FuzzySet> indexFocalElementFuzzySet =
formulatingFocalElement.get(nIndexInput).getTheFuzzySets().iterator();
        while ( indexFocalElementFuzzySet.hasNext() ) {
            // remove all elements up to nIndexInput from  this.fuzzySetsForOneFocalElement to reset for
next iteration
            for (int nRemove = this.fuzzySetsForOneFocalElement.size() - 1; nRemove >= nIndexInput;
nRemove--) {
               if ( Driver.isDebugOn() ) {
                  System.out.println("           RESETTING this.fuzzySetsForOneFocalElement.size(): " +
this.fuzzySetsForOneFocalElement.size() +
                     "   for nIndexInput: " + nIndexInput + .”..Removing input index: " + nRemove);
               }
               this.fuzzySetsForOneFocalElement.remove(nRemove);
            }

            FuzzySet anInputFocalElementFuzzySet = indexFocalElementFuzzySet.next();
            // add to this.fuzzySetsForOneFocalElement
            this.fuzzySetsForOneFocalElement.add(anInputFocalElementFuzzySet);

            // #!#!#!#!##!#!#!#
            // create new focal element for this if nIndexInput is last element of collection
            // #!#!#!#!##!#!#!#
            if (nIndexInput == formulatingFocalElement.size() - 1) {
               // ASSERT  this.fuzzySetsForOneFocalElement has size =
collectionOneFocalElementFromInputs.size()
               // and this.fuzzySetsForOneFocalElement contains fuzzy sets from inputs with
               // index 'n-1' a fuzzy set from input 'n'

               // ADD THE FUZZY SETS TO THE FOCAL ELEMENT TO THIS, AND ADD EVIDENCE
               String nameForCombinedFuzzySets = new String();
               boolean bIsFirstInputFuzzySet = true;
               Iterator<FuzzySet> iterFuzzySet = this.fuzzySetsForOneFocalElement.iterator();
               while (iterFuzzySet.hasNext()) {
                  if (bIsFirstInputFuzzySet) {
                     nameForCombinedFuzzySets += iterFuzzySet.next().getName();
                     bIsFirstInputFuzzySet = false;
                  }
                  // else
                  nameForCombinedFuzzySets += "_&_" + iterFuzzySet.next().getName();
               } // end iterFuzzySet
               this.aFocalElementForThisRuleLinguistic.add(new FuzzySet(nameForCombinedFuzzySets));
               if ( Driver.isDebugOn() ) {
                  System.out.println("           Fuzzy Sets for this.aFocalElementForThisRuleLinguistic: ");
                  for (FuzzySet fuzzSet: this.aFocalElementForThisRuleLinguistic)
                     System.out.println("              " + fuzzSet);
               }

            } // end if (nIndexInput == collectionOneFocalElementFromInputs.size() - 1)

            // recursively continue with the focal elements at the next nIndexInput
            this.setFuzzySetsForOneFocalElement(formulatingFocalElement, nIndexInput + 1);

         } /// end while iterAnInputFocalElements

      } // end setFuzzySetsForOneFocalElement

      // formulatingFocalElement is used by this.formFocalElements() to find combinations of focal elements
of inputs
      private ArrayList<FocalElement> formulatingFocalElement = new ArrayList<FocalElement> ();
```

```
   // fuzzySetsForOneFocalElement is used by this.setFuzzySetsForOneFocalElement() create one focal
element for this
   private ArrayList<FuzzySet> fuzzySetsForOneFocalElement = new ArrayList<FuzzySet> ();

   // aFocalElementForThisRuleLinguistic is used by this.setFuzzySetsForOneFocalElement()
   private ArrayList<FuzzySet> aFocalElementForThisRuleLinguistic = new ArrayList<FuzzySet> ();

   // theFocalElementsForThisRuleLinguistic is used by this.setFuzzySetsForOneFocalElement()
   private ArrayList<FocalElement> theFocalElementsForThisRuleLinguistic = new
ArrayList<FocalElement> ();

   // calculatedDegreeOfEvidence is used by this.setFuzzySetsForOneFocalElement()
   private double calculatedDegreeOfEvidence = 1.0;
```

A.2 Deployment of LinguisticBelief

Using the “Clean and Build Main Project” option in the IDE, a jar file for the LinguisticBelief application was created in the “dist” subdirectory of the project, as indicated in the project file description of Figure A-4. The IDE automatically includes all libraries used by the project in the “dist/lib” subdirectory of the project.

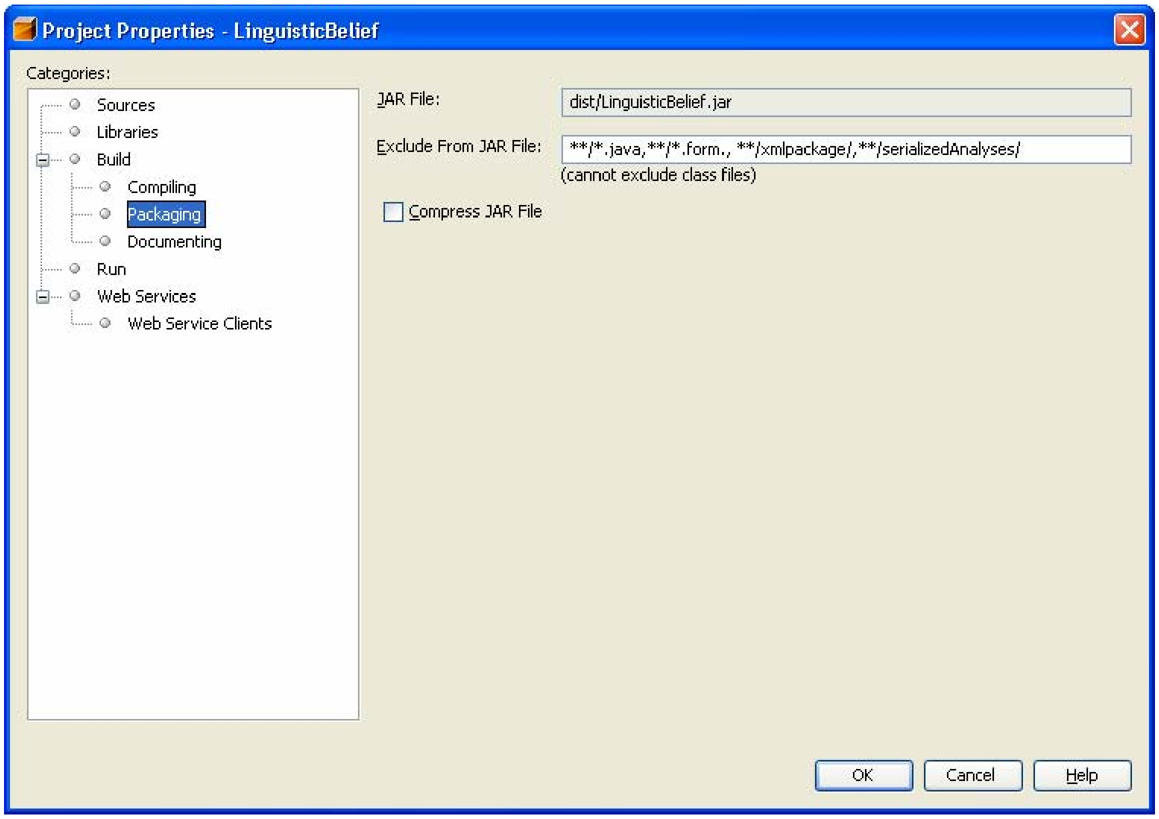

NetBeans IDE 5.0 - LinguisticBelief
File Edit View Navigate Source Refactor Build Run CVS Tools Window Help
Projects Files Runtime
BasicLingBelPlDialog.java BasicLinguisticDialog.java ExceedancePlotDialog.java OrderFuzzySetsDialog.java Driver.java RuleLinguistic.java README.TXT LinguisticBeliefXMLData.dtd Javadoc Index Search
LinguisticBelief
build
dist
lib
jcommon-1.0.6.jar
jfreechart-1.0.3.jar
swing-layout-1.0.jar
LinguisticBelief.jar
META-INF
datapackage
guipackage
images
xmljavatools
README.TXT
images
nbproject
src
test
build.xml
manifest.mf
1:1 INS

```
========================
BUILD OUTPUT DESCRIPTION
========================

When you build an Java application project that has a main class, the IDE
automatically copies all of the JAR
files on the projects classpath to your projects dist/lib folder. The IDE
also adds each of the JAR files to the Class-Path element in the application
JAR files manifest file (MANIFEST.MF).

To run the project from the command line, go to the dist folder and
type the following:

java -jar "LinguisticBelief.jar"

To distribute this project, zip up the dist folder (including the lib folder)
and distribute the ZIP file.

Notes:

* If two JAR files on the project classpath have the same name, only the first
JAR file is copied to the lib folder.
* If the classpath contains a folder of classes or resources, none of the
classpath elements are copied to the dist folder.
* If a library on the projects classpath also has a Class-Path element
specified in the manifest,the content of the Class-Path element has to be on
the projects runtime path.
* To set a main class in a standard Java project, right-click the project node
in the Projects window and choose Properties. Then click Run and enter the
class name in the Main Class field. Alternatively, you can manually type the
class name in the manifest Main-Class element.

```

Finished building LinguisticBelief (clean, jar).

Figure A-4. Deployment Package for LinguisticBelief in the Netbeans IDE

In creating the LinguisticBelief.jar in the IDE, the xmlpackage and the serializedAnalyses packages (shown in Figure A-1) were excluded from the LinguisticBelief.jar file using the properties settings indicated in Figure A-5. These packages were manually added to the distributed package in the "src" subdirectory as shown in Figure A-6. This allows the client direct access to these packages where analyses are saved in xml and serialized form, respectively.



JAR File: dist/LinguisticBelief.jar

Exclude From JAR File: **/*.java,**/*.form., **/xmlpackage/,**/serializedAnalyses/

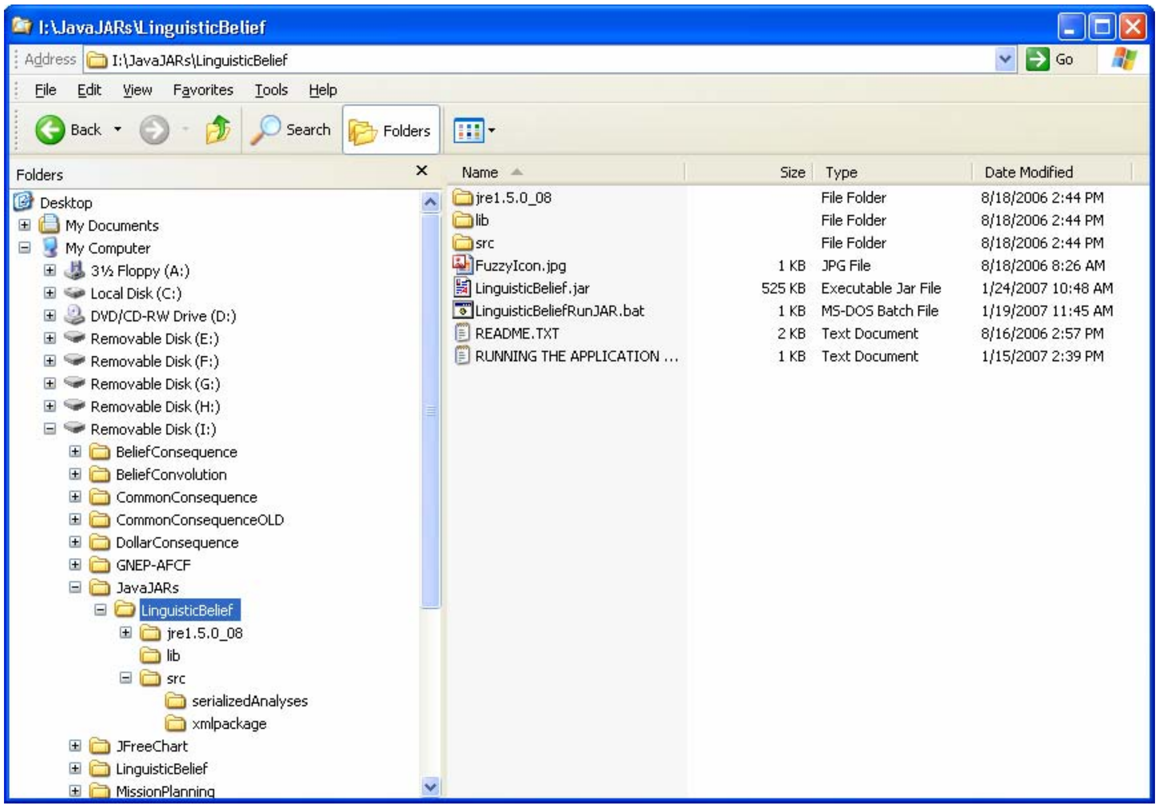
(cannot exclude class files)

☐ Compress JAR File

Figure A-5. Setting Properties in the IDE to Not Jar the xmlpackage and serializedAnalyses Packages

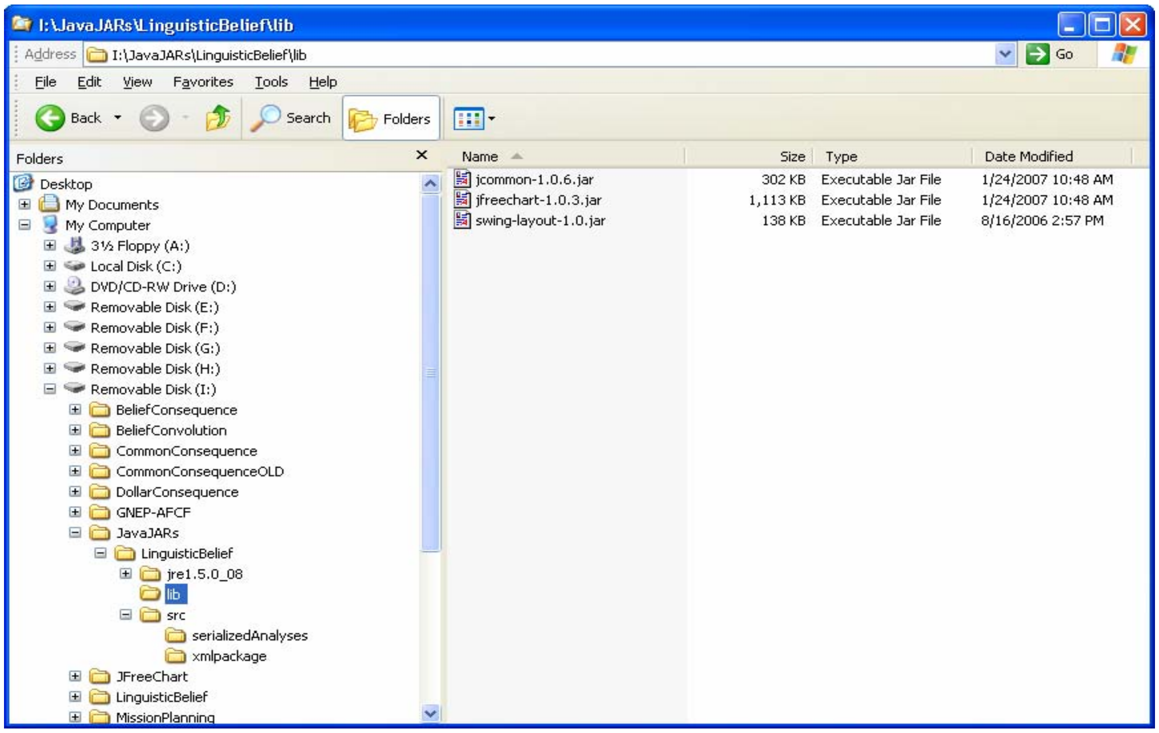
The deployed project is distributed under the name LinguisiticBelief as indicated in Figure A-6. A recent Java runtime environment (jre1.5.0_08) is included in the deployment, so that the user will have JDK5 available as required to run the application. A batch file, LinguisticBeliefRunJAR.bat, is included for running the application in Windows XP using the deployed Java runtime environment.

Figure A-7 shows the files in the lib subdirectory of the deployment package.



Name	Size	Type	Date Modified
jre1.5.0_08		File Folder	8/18/2006 2:44 PM
lib		File Folder	8/18/2006 2:44 PM
src		File Folder	8/18/2006 2:44 PM
FuzzyIcon.jpg	1 KB	JPG File	8/18/2006 8:26 AM
LinguisticBelief.jar	525 KB	Executable Jar File	1/24/2007 10:48 AM
LinguisticBeliefRunJAR.bat	1 KB	MS-DOS Batch File	1/19/2007 11:45 AM
README.TXT	2 KB	Text Document	8/16/2006 2:57 PM
RUNNING THE APPLICATION ...	1 KB	Text Document	1/15/2007 2:39 PM

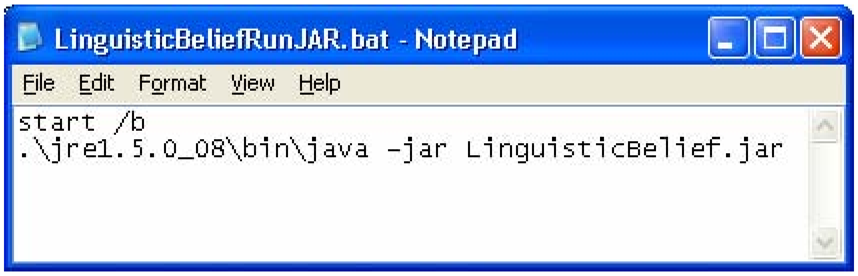
Figure A-6. The LinguisticBelief Deployment Package



Name	Size	Type	Date Modified
jcommon-1.0.6.jar	302 KB	Executable Jar File	1/24/2007 10:48 AM
jfreechart-1.0.3.jar	1,113 KB	Executable Jar File	1/24/2007 10:48 AM
swing-layout-1.0.jar	138 KB	Executable Jar File	8/16/2006 2:57 PM

Figure A-7. Contents of LinguisticBelief.lib

If this package is copied to a client machine, the application can be executed by running the "LinguisticBeliefRunJAR.bat" file. The commands in "LinguisticBeliefRunJAR.bat" are shown in Figure A-8.


LinguisticBeliefRunJAR.bat - Notepad
File Edit Format View Help

```
start /b
.\jre1.5.0_08\bin\java -jar LinguisticBelief.jar
```

Figure A-8. Commands in LinguisticBeliefRunJAR.bat

The distributed package contains the xmlpackage and serializedAnalyses subdirectories as indicated in Figure A-6. The xmlpackage contains the document type description (DTD) file for the client-generated xml files, LinguisticBeliefXMLData.dtd, which must **not** be deleted from the distributed package.

References

Gilbert, D., "The JFreeChart Class Library, Version 1.0.3, Developer's Guide," Object Refinery Limited, January 12, 2007.

Darby, J., "Evaluation of Risk from Acts of Terrorism: The Adversary/Defender Model Using Belief and Fuzzy Sets," SAND2006-5777, Sandia National Laboratories, Albuquerque, NM, September, 2006.

Distribution

3	MS 0757	John Darby, 06462
1	MS 0762	S. Jordan, 06407
2	MS 9018	Central Technical Files, 08944
2	MS 0899	Technical Library, 04536