



Optimizing Customer Loyalty

An Integrated Market Research Workflow Using Bayesian Networks and BayesiaLab

Stefan Conrady, Managing Partner, Bayesia USA

Dr. Lionel Jouffe, CEO, Bayesia

February 14, 2014

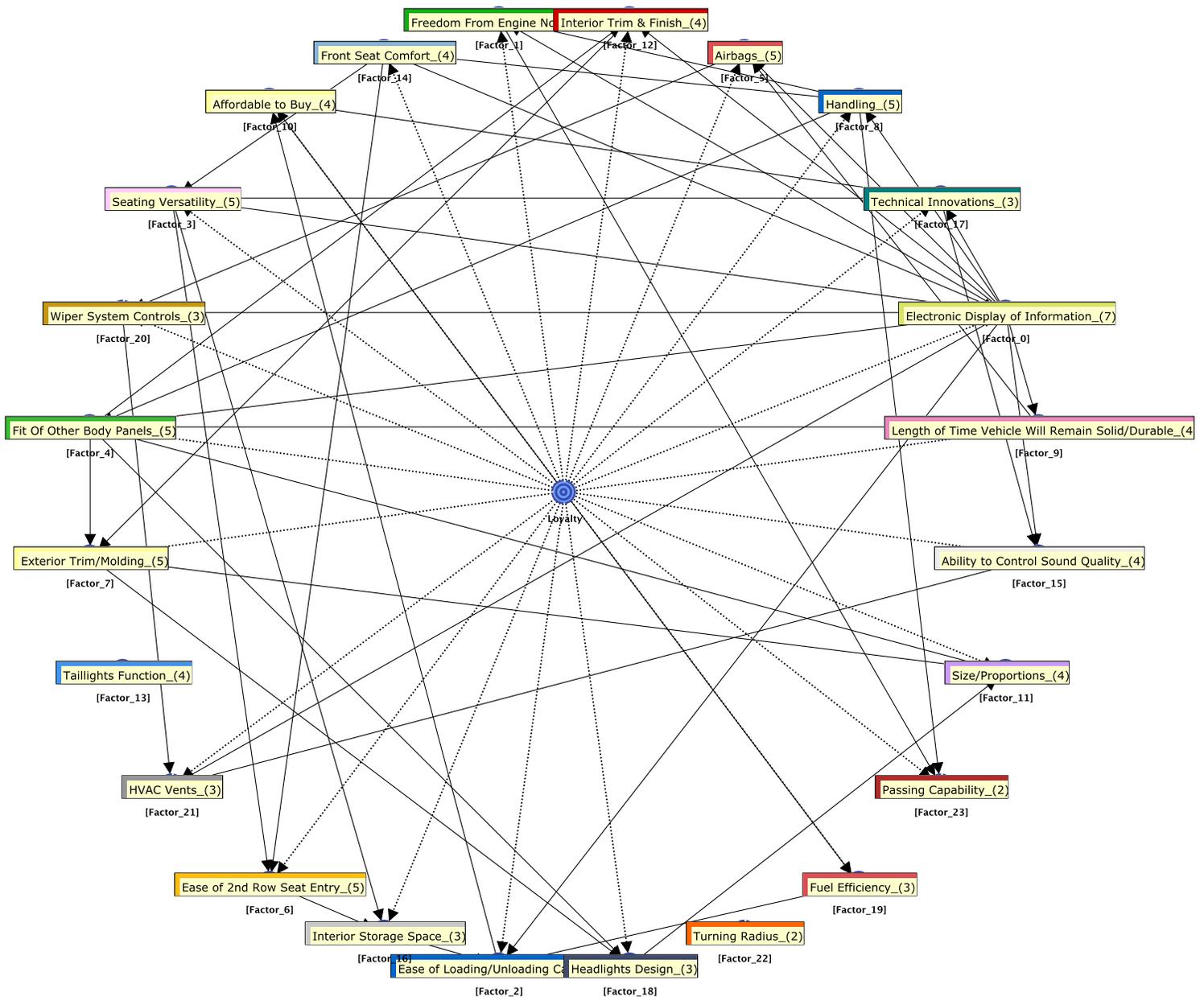


Table of Contents

Executive Summary

Identifying Priorities for Maximizing Repurchase Intent	4
Challenge: Indistinguishable Drivers of Loyalty	4
Solution: Bayesian Networks as Modeling Framework	4
Implementation	5
Other Benefits	5

Introduction

Background	7
Workflow Overview	8
Latent Factor Induction	8
Multi-Level Analysis	8
Optimization	9
Acknowledgements	10
Notation	10

Tutorial

Source Data	11
Data Selection	11
Coding	11
Data Import	12
Unsupervised Structural Learning	18
Mapping	24
Missing Values Imputation	25
Variable Clustering	27
Latent Factor Induction via Multiple Clustering	33
Factor States/Values	36
Introducing the Target Node	38

Focusing on Factors	40
Supervised Learning	43
Structural Coefficient Analysis	44
Target Mean Analysis	46
Multi-Quadrant Analysis (Total Market → Segment)	50
Full-Size Pickup Segment	55
Multi-Quadrant Analysis (Segment → Model)	60
Relearning the Structure at the Market Level	61
Variations within the Realm of the Possible	69
Optimization	71
Recommendation for GMC Sierra 1500	78
Summary	80
Appendix	
Variables	81
List of Factors	82
Contact Information	

Executive Summary

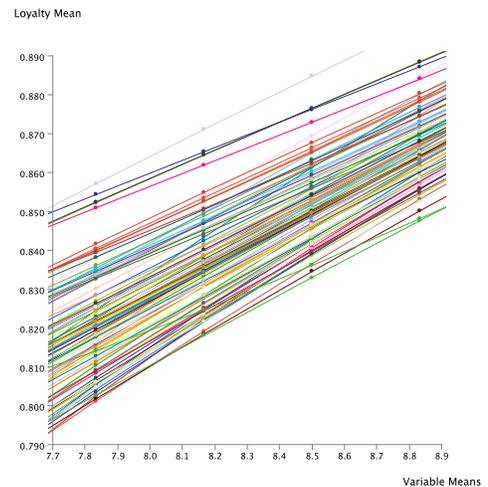
Identifying Priorities for Maximizing Repurchase Intent

This tutorial illustrates an innovative market research workflow for deriving marketing and product planning priorities from auto buyer surveys. In this study, we utilize the Strategic Vision New Vehicle Experience Survey, which includes, among many other items, customers' satisfaction ratings with regard to over 100 individual product attributes.

Challenge: Indistinguishable Drivers of Loyalty

With traditional statistical methods, it has been difficult to rank the importance of individual product attribute ratings with regard to an overall measure, such as repurchase loyalty.

The key challenge is that customers' ratings of individual product attributes are highly correlated. When plotted, we see 100 lines that are nearly indistinguishable in terms of their slope. Given this collinearity of all variables, traditional statistical methods fail to distinguish the importance of individual ratings. We could only naively conclude that an improvement in any rating would generally be associated with higher loyalty. No clear priorities could be established on such a basis.



Solution: Bayesian Networks as Modeling Framework

To overcome this problem we employ an alternative framework: we use Bayesian networks as the mathematical formalism, plus the machine-learning and optimization algorithms of the BayesiaLab software package. This approach embraces collinearity as a feature in the model, instead of suppressing it as a nuisance.

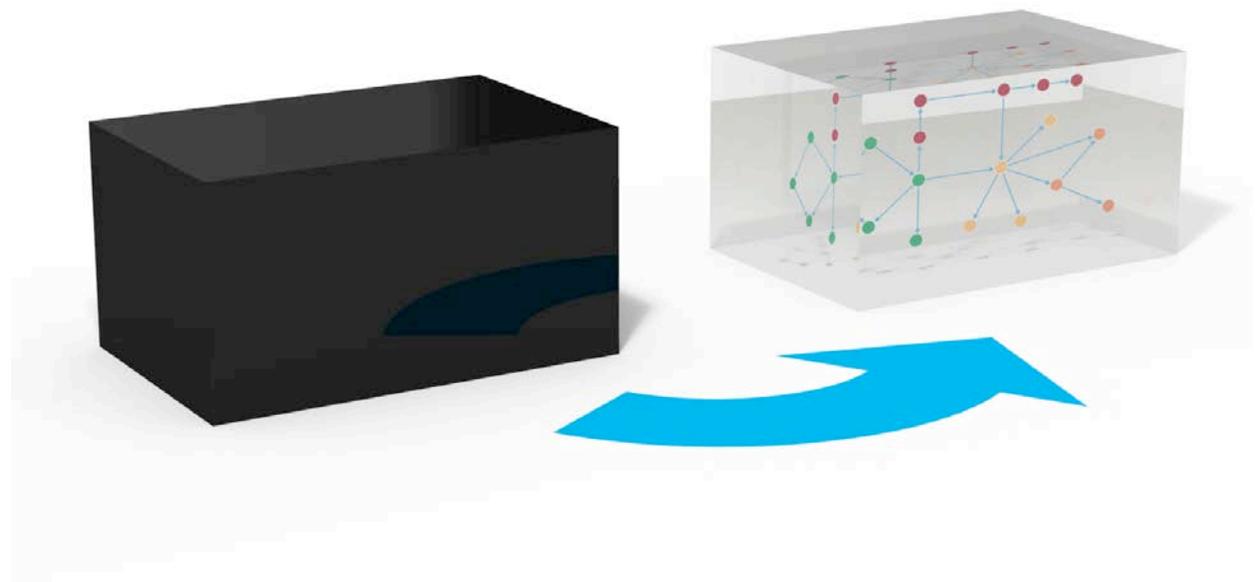
Implementation

First, using BayesiaLab, we machine-learn a Bayesian network that models customers' brand loyalty as a function of their ratings of their current vehicle. This identifies key factors as loyalty drivers in the overall market, at the segment level, and finally at the model level. With these factors identified, we perform optimization, for each vehicle within its competitive context. As a result, we obtain a list of specific priorities for each vehicle, along with the simulated gain in loyalty.

Other Benefits

No Black Box

Many modeling techniques offered in the field of marketing science are opaque to the end user of the research. The nature of many models make them inherently black-box, and thus require a leap of faith by the decision maker.



Not so in our research framework with Bayesian networks. Regardless of one's quantitative skills, any subject matter expert can—by simply using common sense—interpret the Bayesian network models generated with our workflow. Any stakeholder can immediately scrutinize such a model, thus enabling him to verify its structure, or, by using his domain knowledge, to invalidate it. Their inherent falsifiability makes Bayesian networks ideal scientific tools.

Real-Time Recommendations

In most organizations, waiting for research results and their interpretation is a matter of months. The time span between a consumer sentiment expressed in a survey, and a company's response, can sometimes even exceed the lifecycle length of a product.

Our workflow creates a single, direct, and transparent link from data to recommendation. This directness provides unprecedented analysis speed. We reduce the lag between receipt of data and delivery of recommendation from months to days. As a result, near real-time policy recommendations are feasible for the first time.

Introduction

Background

Market Maturity and Homogeneity

The auto industry is an example of a very mature market. It is fair to say that all automakers offer high-quality products these days in North America. The proverbial “lemons” are few and far between. Fierce competition has led to product offerings that are remarkably similar for their respective vehicle category, both in their specifications and their functional performance. With similar cost and budget constraints, and an overlapping supplier base for all manufacturers, the auto business is mostly about eking out minute advantages, as opposed creating fundamental breakthroughs.

No doubt, the brand plays a major role in buying decisions. Hence, marketing, branding and promotion efforts of automakers typically absorb a similar amount of resources as the actual R&D expenses for vehicle development. For the purpose of this paper, however, we will not venture into the challenging domain of return on marketing investment. This is a topic for another methodology tutorial in the future.

Given this overall quality and performance homogeneity, consumer perceptions—as we will see in this study—are also remarkably homogenous across similar kinds of vehicles. For market researchers, it is thus very difficult to “tease out” material differences in customers’ perception of product attributes of competitive vehicles. It is even more challenging to establish which of these similarly-perceived vehicle characteristics do really matter when it comes to buying an automobile.

Loyalty

“It is cheaper to keep a customer than to find a new one” is an often-quoted marketing adage. Loyalty is a very relevant quantity, much more tangible than mere satisfaction. Given the maturity of the auto market and rather lengthy ownership cycles, repurchase loyalty is of special significance. Thus, we go beyond satisfaction in this study, and rather link product ratings to stated repurchase intent.

We will focus exclusively on how customers’ product ratings can affect loyalty, i.e. find “what really matters” when it come to brand loyalty. We will present a methodology that can identify the relevance of minute differences in consumer perception, in order to help identify priorities among a plethora of opportunities to improve product ratings.

Workflow Overview

Latent Factor Induction

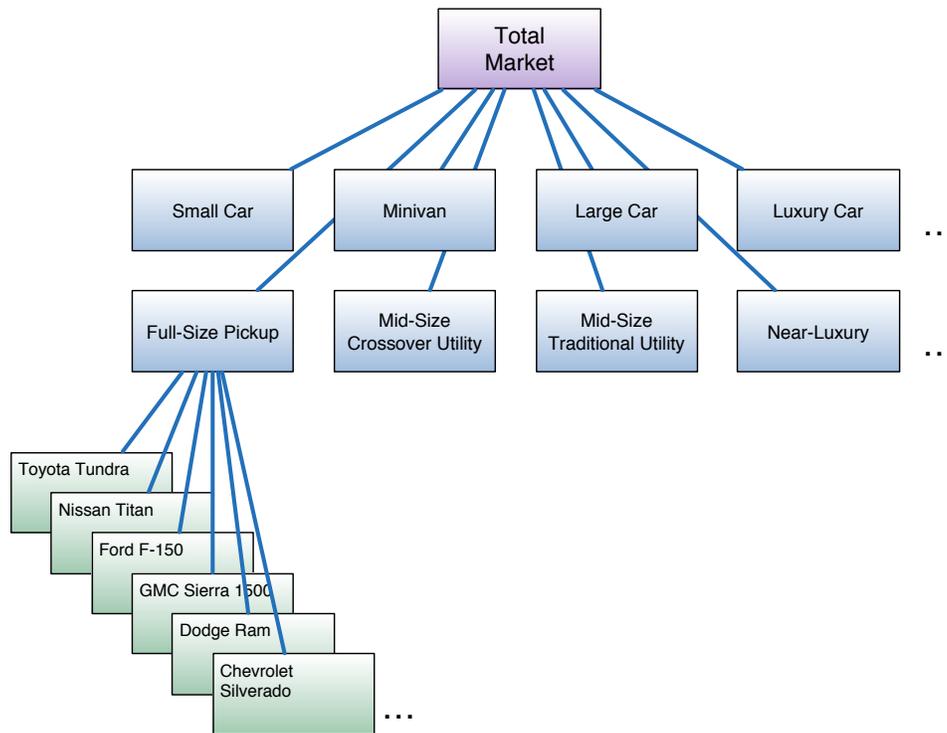
In this paper, we employ BayesiaLab's machine learning algorithms to generate Bayesian networks that will allow us to identify major concepts, i.e. latent factors, from the observed satisfaction ratings, i.e. manifest variables. Inducing factors creates a level of abstraction that will allow us to see a "bigger picture," that is more stable than if it is only based on manifest variables. Once factors are identified, we will examine how they "drive" brand loyalty. Ultimately, we want to establish the effect of these factors with regard to the outcome variable, i.e. loyalty.

Multi-Level Analysis

We will examine the loyalty drivers at multiple levels of the market. We will identify the general areas of opportunity for loyalty improvement at the segment¹ level and at the vehicle model² level.

¹ "Segment" refers to a vehicle category, such as *Subcompacts* or *Large Sedans*. There are numerous segmentation schemes used in the auto industry, which all rely on custom terminology. However, all automakers agree on the definition of the Full-Size Pickup segment, on which we focus in this study.

² "Vehicle model" refers to a vehicle make (brand) and model/line, e.g. Ford Explorer, Nissan Altima. In this case study, we will not drill down to the vehicle trim level, e.g. Ford Explorer Limited and Nissan Altima 2.5 S.



More specifically, we will proceed from the overall market to the *Full-Size Pickup* segment, and then to the vehicle models within it. We chose this particular segment primarily for expository simplicity. It is a very well defined segment in terms of vehicle characteristics while only consisting of a few major contenders. Plus, it is one of the most important segments in the U.S. auto industry, both in terms of volume and profitability.

Optimization

Once loyalty drivers are modeled, we will identify priorities for improvements by vehicle model. For each model, in its specific competitive context, our approach will generate recommendations with the objective of improving brand loyalty.

As we examine the impact of satisfaction ratings on loyalty, we need to remember though that satisfaction ratings are inherently subjective. The recommendations we will present do not necessarily specify the means by which ratings should be improved.

Acknowledgements

We would like to express our gratitude to Alexander Edwards, President of Strategic Vision, Inc.³, for generously providing data from their 2009 New Vehicle Experience Survey for our case study.

Notation

To clearly distinguish between natural language, software-specific functions and example-specific variable names, the following notation is used:

- Bayesian network and BayesiaLab-specific functions, keywords, commands, etc., are capitalized and shown in bold type.
- Names of attributes, variables, nodes and names of node states are italicized.

³ Strategic Vision is a research-based consultancy with more than 35 years of experience in understanding the consumers' and constituents' decision-making systems for a variety of Fortune 100 clients, 10 Downing Street, Coca-Cola, American Airlines, Proctor & Gamble, the White House and including most automotive manufacturers and many advertising agencies. The company specializes in identifying consumers' complete, motivational hierarchies, including the product attributes, personal benefits, value/emotions and images that drive perceptions and behaviors. Strategic Vision has at its core a large-scale syndicated automotive experience and "Pulse of the Customer" (POC) study that collects more than 350,000 responses annually, using over 1,500 comprehensive data points. Since its foundation in 1972 and incorporation in 1989, Strategic Vision—led by company founders Darrel Edwards, Ph.D., J. Susan Johnson, Sharon Shedroff, with Alexander Edwards—has used in-depth Discovery Interviews and Value Centered Survey instruments that provide comprehensive, integrated and actionable outcomes, linking behavior to attributes to consequences to values and emotions to images (www.strategicvision.com).

Tutorial

Source Data

Our case study uses real-world data from the auto industry, which has conducted customer satisfaction research for decades. More specifically, we utilize the 2009 New Vehicle Experience Survey (NVES), a syndicated study conducted by Strategic Vision, Inc., which surveys new vehicle buyers in the U.S. This study is widely used in the auto industry, and it is one of principal resources for market researchers and product planners. NVES contains over 1,000 variables and close to 200,000 respondent records. Among many demographic and psychographic variables, NVES contains 98 individual satisfaction measures, ranging from *Acceleration* to *Wiper System Controls*.

Data Selection

From the original NVES dataset consisting of 1,089 columns and 71,200 rows, we select 103 columns that are relevant for the purposes of this tutorial.⁴

The first group of columns refers to the vehicle type, e.g. make, model and vehicle segment. The second group includes 98 columns that all concern the vehicle buyer's satisfaction with specific aspects of the purchased vehicle, ranging from *Ability to Control Sound* to *Wiper System Controls*.

For notational convenience, we rename a number of frequently used variables as follows:

- *New Model Segment* → *Segment*
- *New Model Purchased - Brand* → *Make*
- *New Model Purchased (Alpha Order)* → *Make/Model*
- *Rate Buy Another From Same Manufac.* → *Loyalty*

Coding

NVES measures all satisfaction-related variables on an ordinal scale, from 1 (*A failure*) to 5 (*Delightful*), as shown in the excerpt from the printed questionnaire.

⁴ See the appendix for a complete list of the selected variables.

Your satisfaction with specific aspects of your new vehicle . . .

1. Please use the following scale and “X” your ratings of your new vehicle on the features listed below. The scale ranges from the highest rating “Delightful” to the lowest rating “A failure”.

Delightful	Excellent	Satisfactory	Unsatisfactory	A failure
5	4	3	2	1

<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%;"></td> <td style="text-align: center;">Delightful</td> <td style="text-align: center;">A failure</td> </tr> <tr> <td>Overall exterior styling</td> <td style="text-align: center;">5 4 3 2 1</td> <td style="text-align: center;">5 4 3 2 1</td> </tr> <tr> <td>Lines/flow of the vehicle</td> <td style="text-align: center;">5 4 3 2 1</td> <td style="text-align: center;">5 4 3 2 1</td> </tr> <tr> <td>Paint job appearance</td> <td style="text-align: center;">5 4 3 2 1</td> <td style="text-align: center;">5 4 3 2 1</td> </tr> </table>		Delightful	A failure	Overall exterior styling	5 4 3 2 1	5 4 3 2 1	Lines/flow of the vehicle	5 4 3 2 1	5 4 3 2 1	Paint job appearance	5 4 3 2 1	5 4 3 2 1	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%;"></td> <td style="text-align: center;">Delightful</td> <td style="text-align: center;">A failure</td> </tr> <tr> <td>Overall interior styling</td> <td style="text-align: center;">5 4 3 2 1</td> <td style="text-align: center;">5 4 3 2 1</td> </tr> <tr> <td>Interior colors</td> <td style="text-align: center;">5 4 3 2 1</td> <td style="text-align: center;">5 4 3 2 1</td> </tr> <tr> <td>Interior trim and finish</td> <td style="text-align: center;">5 4 3 2 1</td> <td style="text-align: center;">5 4 3 2 1</td> </tr> </table>		Delightful	A failure	Overall interior styling	5 4 3 2 1	5 4 3 2 1	Interior colors	5 4 3 2 1	5 4 3 2 1	Interior trim and finish	5 4 3 2 1	5 4 3 2 1
	Delightful	A failure																							
Overall exterior styling	5 4 3 2 1	5 4 3 2 1																							
Lines/flow of the vehicle	5 4 3 2 1	5 4 3 2 1																							
Paint job appearance	5 4 3 2 1	5 4 3 2 1																							
	Delightful	A failure																							
Overall interior styling	5 4 3 2 1	5 4 3 2 1																							
Interior colors	5 4 3 2 1	5 4 3 2 1																							
Interior trim and finish	5 4 3 2 1	5 4 3 2 1																							

For analysis purposes, most market researchers have typically been using the NVES satisfaction ratings linearly transformed into a 1.5-9.5 scale. We will follow this convention in our tutorial.

Loyalty is asked in the NVES with the following question, which also features an ordinal scale for the response, ranging from *Definitely will not* to *Definitely will*.

6. Everything considered, how likely are you in the future to purchase or lease another new vehicle made by the same manufacturer ?

Definitely will
 Probably will
 Don't know
 Probably will not
 Definitely will not

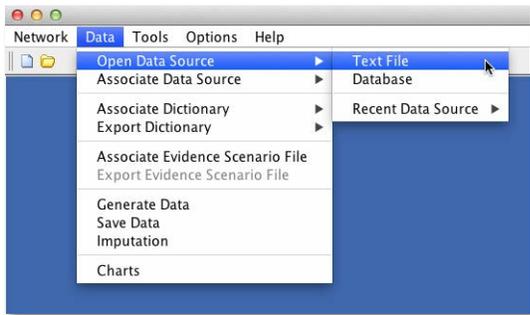
Given that our objective is loyalty optimization, we need to associate a numerical value with each of the ordinal states of this response variable. The following linear assignment of probabilities is somewhat arbitrary, but for the purpose of this study we will accept it as a reasonable approximation.

Response	Probability
Definitely Will Not	0.00
Probably Will Not	0.25
Do Not Know	0.50
Probably Will	0.75
Definitely Will	1.00

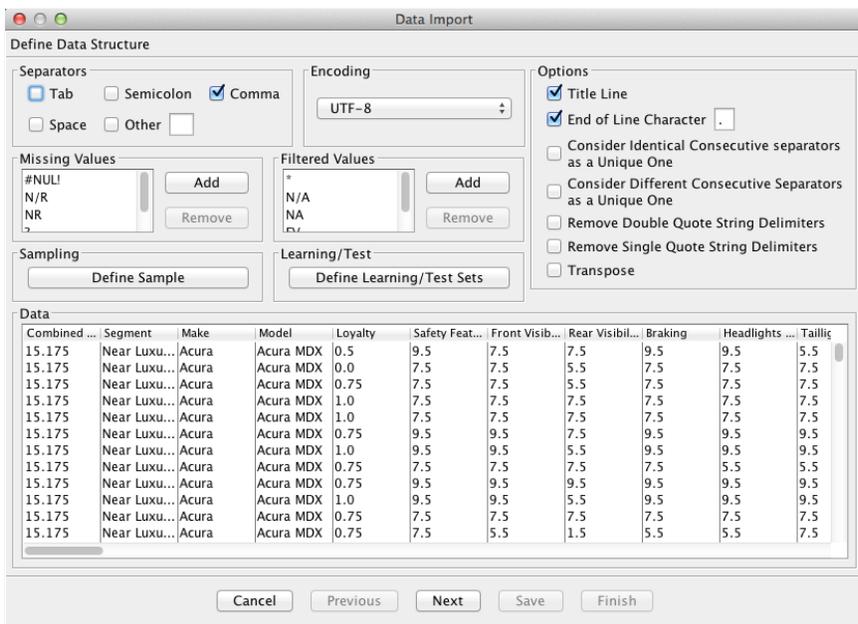
Data Import

To start the analysis with BayesiaLab, we first import the survey data set, which was provided as a CSV file.⁵ With **Data | Open Data Source | Text File**, we start the **Data Import Wizard**, which immediately provides a preview of the data file.

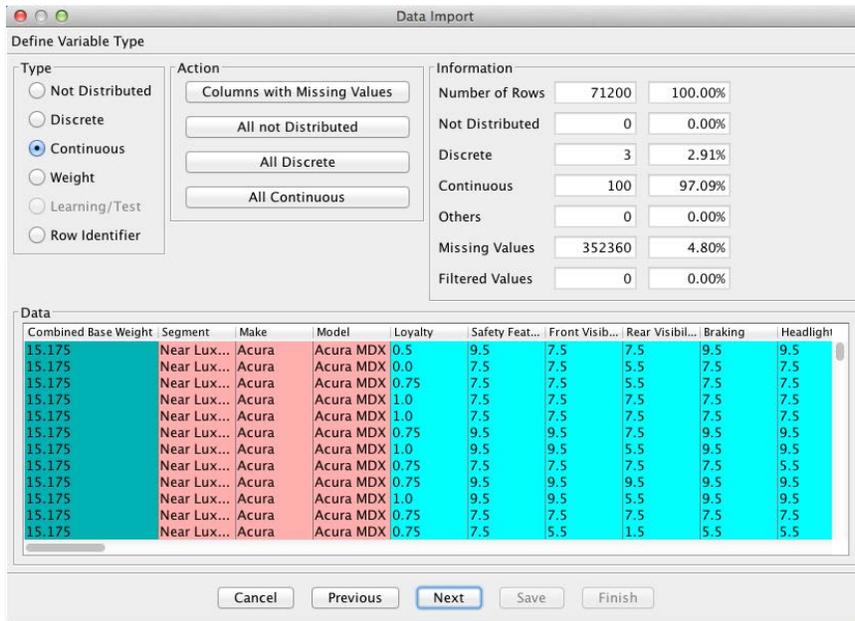
⁵ CSV stands for “comma-separated values”, a common format for text-based data files.



The table displayed in the **Data Import** wizard shows the individual variables as columns and the responses as rows. There are a number of options available, e.g. for sampling. However, this is not necessary in our example given the relatively small size of the database.

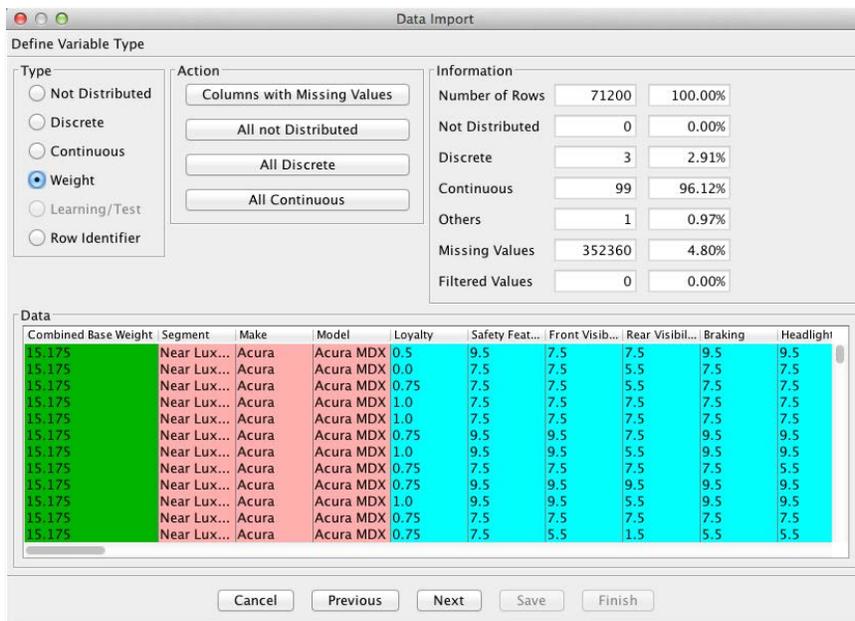


Clicking the **Next** button, prompts a data type analysis, which provides BayesiaLab's best guess regarding the data type of each variable.



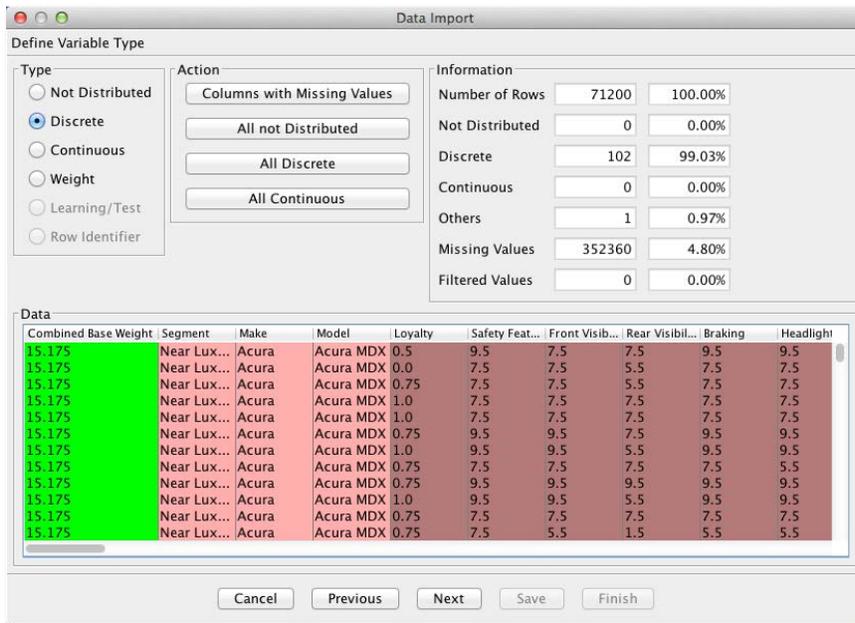
Furthermore, the **Information** box provides a brief summary regarding the number of records, the number of missing values ⁶, filtered states, etc.

In this example, we will need to override the default data type for the *Combined Base Rate* variable. This variable serves as the survey weight of each observation. We change the data type by highlighting the column and clicking the **Weight** check box, which changes the color of the *Combined Base Rate* column to green.



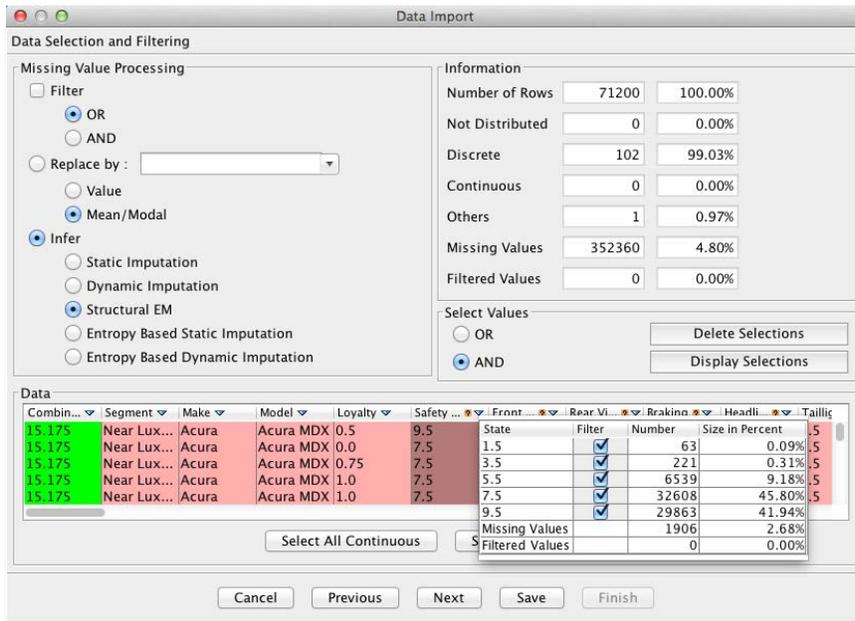
⁶ There are no missing values in our database and filtered states are not applicable in this survey.

BayesiaLab interprets all the numerical columns as **Continuous**, which is technically correct. Consequently, BayesiaLab would attempt to discretize these variables. However, in our case, these variables were already discretized by the response levels given in the questionnaire. Thus, we will use this discretization as is. We do so by highlighting all continuous variables and then ticking the **Discrete** checkbox.



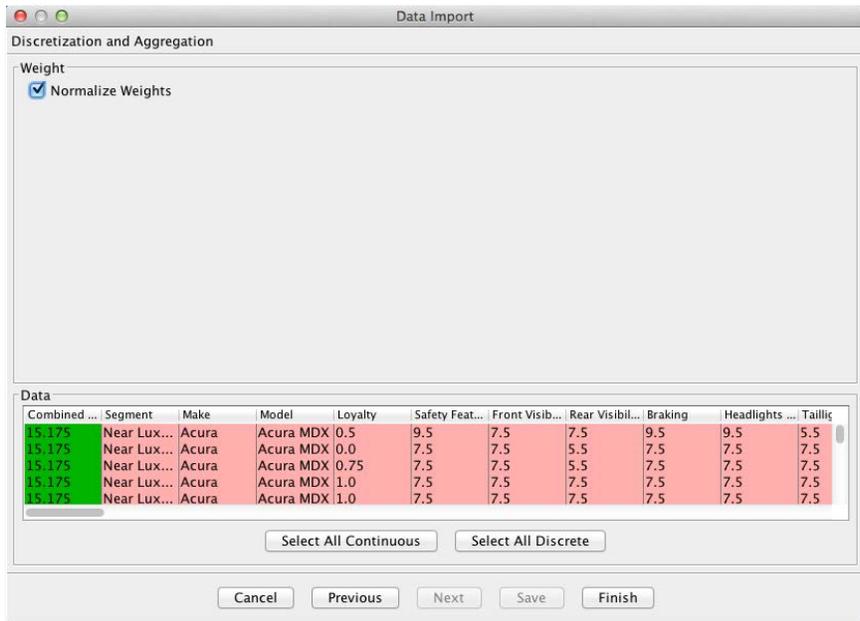
The next screen provides options as to how to treat any missing values. As shown by the **Information** panel, 4.8% of all values are missing in this dataset. Clicking the small upside-down triangle next to the variable names brings up a window with key statistics of the selected variable, in this case *Loyalty*.

Furthermore, columns with missing values are highlighted with a small question mark symbol. By highlighting any such variable, we are given the option of selecting the missing values imputation algorithm under the **Infer** section of the window.



Given that we are using weights in this dataset, we now have an option to **Normalize Weights**. If we left this option unchecked, each record in the dataset would be counted as many times as the weight indicates. For instance, the first row would be counted 15.175 times. Applied to all rows, this would yield the correct proportion of observations relative to each other. However, BayesiaLab would subsequently end up “over-learning” from 3,233,840.85 weighted observations.⁷ To avoid generating a false sense of precision, we select **Normalize Weights**, so each response, on average, represents one transaction.

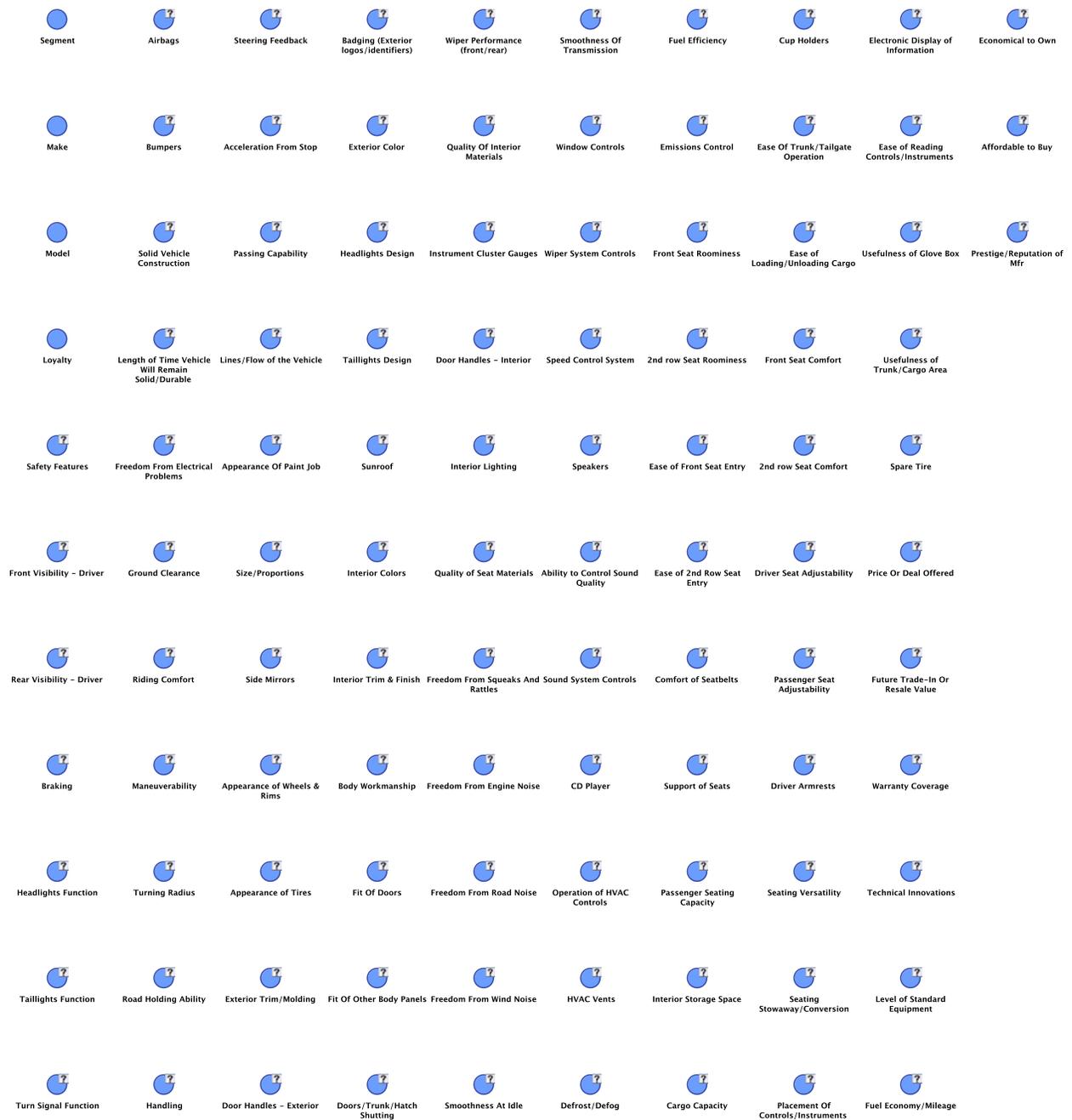
⁷ Each response in this survey, on average, represents roughly 45 vehicle purchases.



Upon completion of the import process, we obtain an initially unconnected network, which is shown below in the screenshot. All variables are now represented as nodes, one of the core building blocks in a Bayesian network. A node can stand for any variable of interest. Once the variables appear in this form in a graph, we will exclusively refer to them as nodes.

The wide spectrum of nodes can now be seen at a glance. For clarity, this network is shown without its graph panel window. Whenever the context is clear, we will present the network by itself in this tutorial.

Optimizing Customer Loyalty

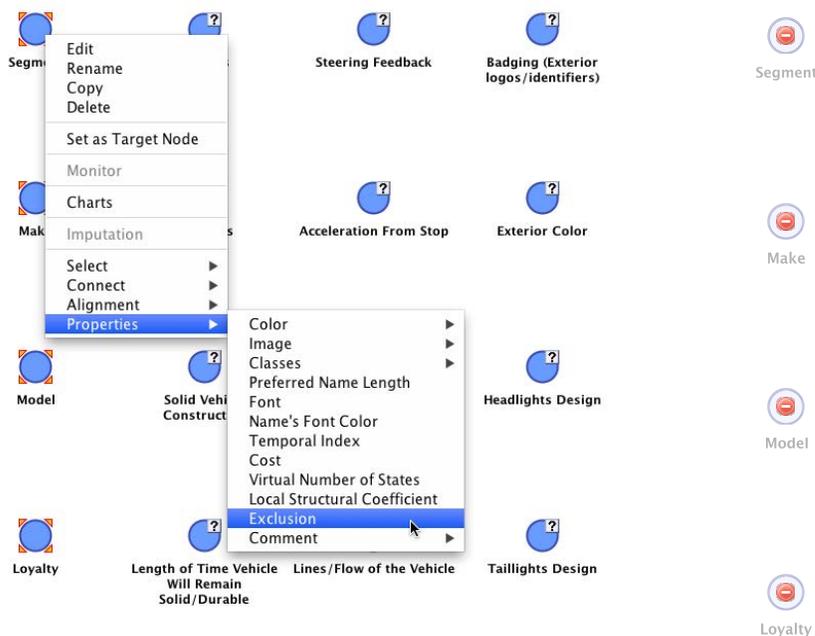


Unsupervised Structural Learning

A central element in our study is to look for overarching concepts among the 98 satisfaction measures in our data set. Once identified, these concepts will subsequently serve as factors as we further develop our model.

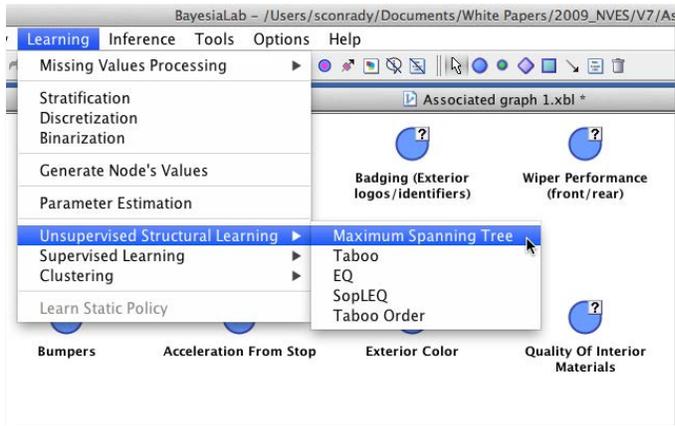
Machine learning a Bayesian network is a remarkably practical way to identify easily-interpretable variable clusters for factor induction. Among BayesiaLab's **Unsupervised Learning** algorithms, the **Maximum Weight Spanning Tree** is a very efficient approach for quickly obtaining a Bayesian network that can provide the basis for variable clustering. The speed of this particular method is due to a key constraint, namely that in the to-be-learned network, each node is restricted to having only one parent node. This massively reduces the number of candidate networks that the learning algorithm must examine.

Prior to initiating the clustering process, we need to exclude any nodes that we do not want to be included in the clustering, such as the node that we will later use as target node and those that we will use as breakout variables, e.g. *Segment*. We exclude them by right-clicking on any of the selected nodes and then clicking **Properties | Exclusion** from the contextual menu. Alternatively, holding “x” while double-clicking the nodes performs the same exclusion function. Once excluded, the nodes will appear as .

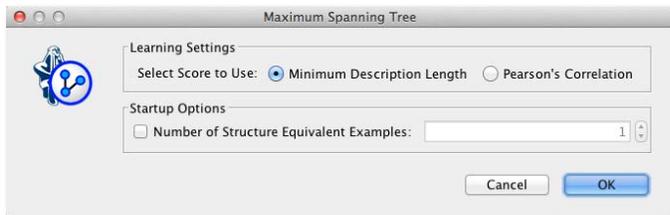


Now we can start the learning process from the main menu: **Learning | Unsupervised Structural Learning | Maximum Spanning Tree**.

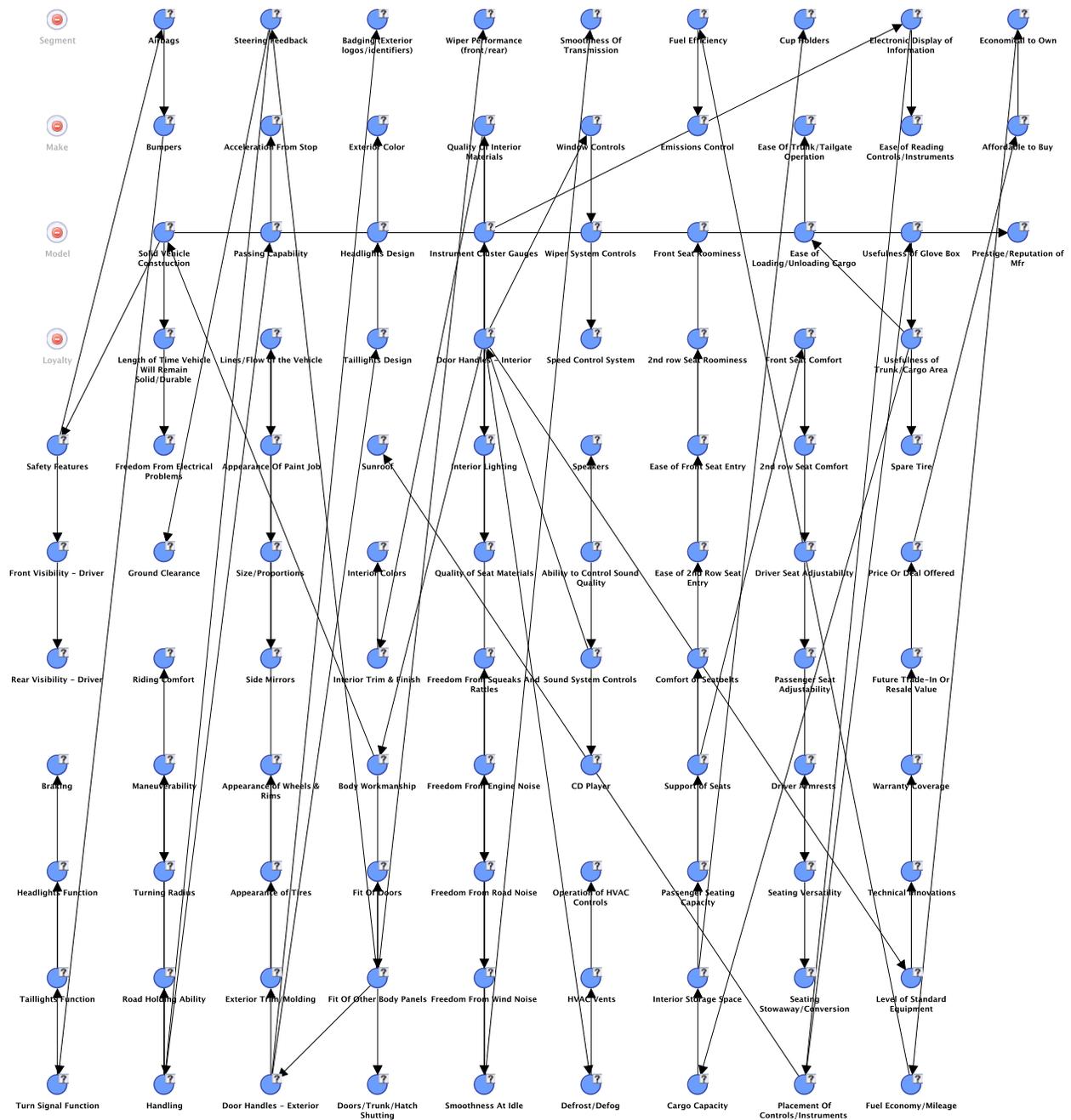
Optimizing Customer Loyalty



The **Maximum Weight Spanning Tree** is the only learning algorithm in BayesiaLab that offers an alternative to the **Minimum Description Length (MDL)** as the learning score. However, without further explanation, we will stick to the default and confirm **MDL**.

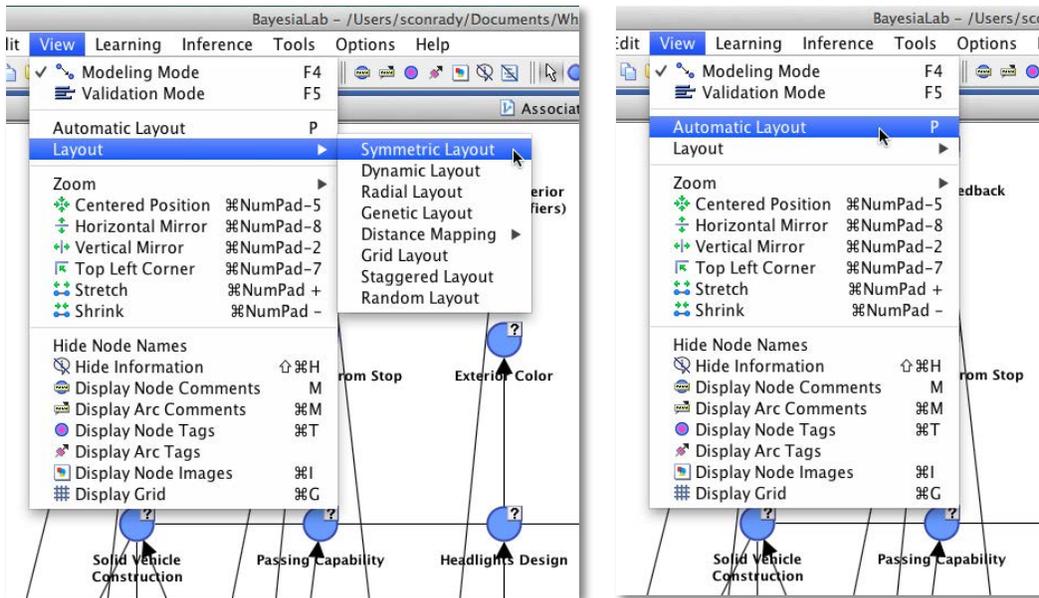


This default view of the resulting network is hardly intuitive. Hence, throughout this exercise, we will make frequent use of one of several available layout algorithms.

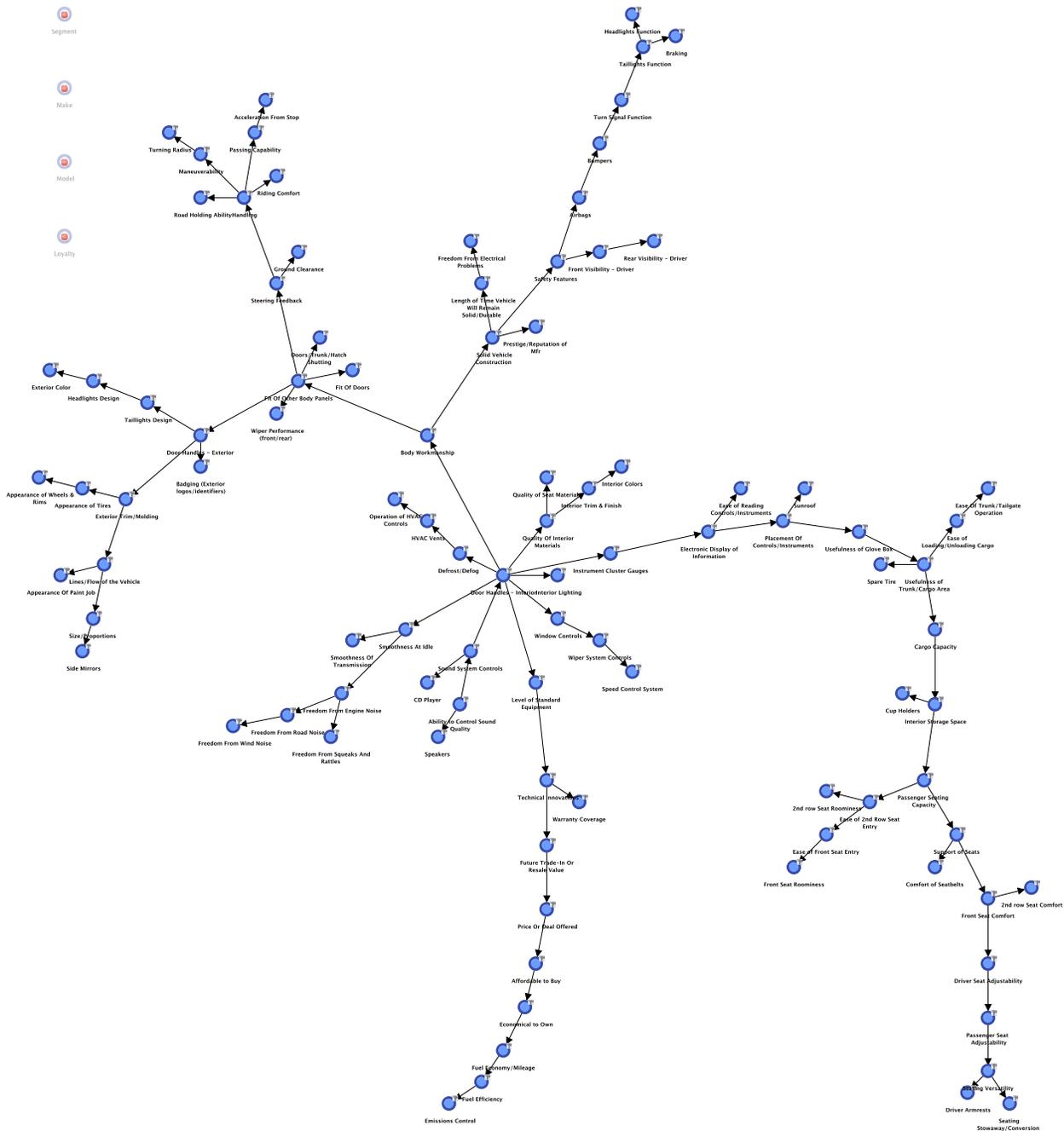


All layout algorithms are accessible from the main menu via **View | Layout**. Most often, we will use **View | Automatic Layout**, or alternatively use the shortcut “P”.

Optimizing Customer Loyalty



Upon applying a layout algorithm, we obtain an untangled version of the original network. Note that the structure of the network remains unchanged. We can further adjust the positions of the nodes as needed to create a legible and interpretable layout.



In this network, we can examine the probabilistic relationships between the nodes, which are represented as arcs. The structure of the network lends itself to perform a “sanity check” versus our own domain knowledge. For instance, we can look at the lower right branch of the network and see that all these nodes relate to the vehicle interior and seating. It should not surprise us to see that *Front Seat Roominess* is directly connected to *Ease of Front Seat Entry*.

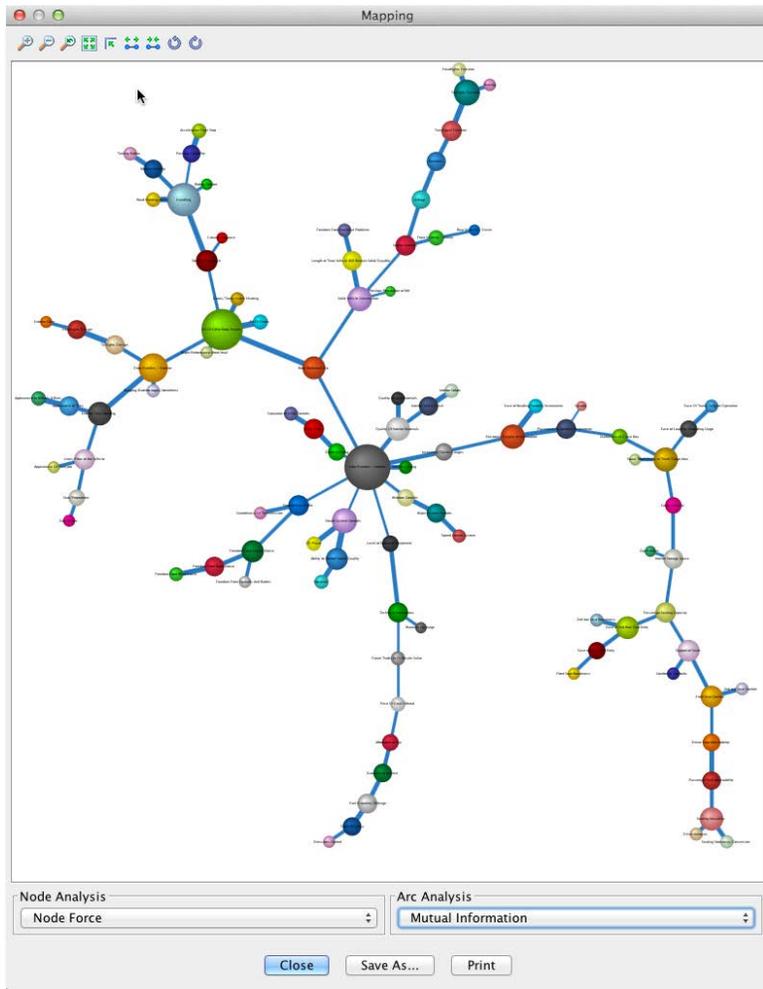
Mapping

Beyond this kind of qualitative assessment, BayesiaLab's **Mapping** function is very helpful for interpreting the importance of the nodes and the strength of the relationships between them.

We can initiate **Mapping** from within the **Validation Mode** by selecting **Analysis | Visual | Mapping**.



The **Mapping** window features drop-down menus for **Node Analysis** and **Arc Analysis**. We select **Node Force** for **Node Analysis**; for **Arc Analysis** with choose **Mutual Information**.



In the resulting graph, as per our selection, the node size reflects the **Node Force**, the arc thickness reflects the **Mutual Information**. This visualization suggests that *Solid Vehicle Construction* and *Handling* are among the most important nodes. All measures appear plausible, so we feel comfortable to move forward on this basis.

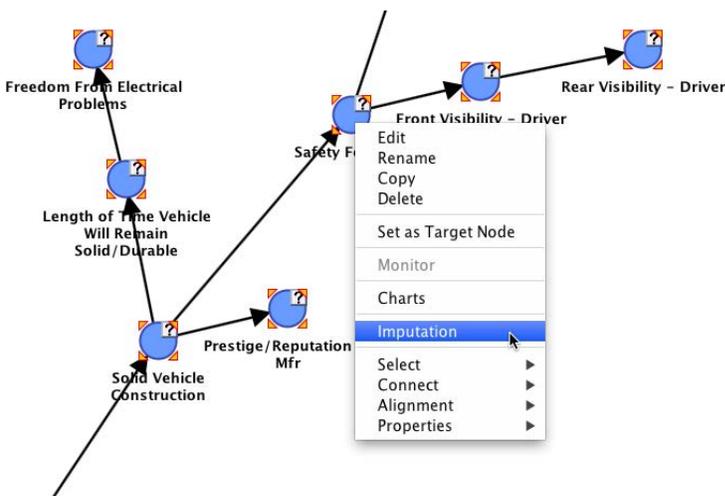
Missing Values Imputation

As we already saw during the data import process, all rating variables contain missing values. Given that we set **Structural EM** as the missing values imputation algorithm, we could proceed with our entire analysis without further thinking about the missing values. The **Structural EM** algorithm would handle them auto-

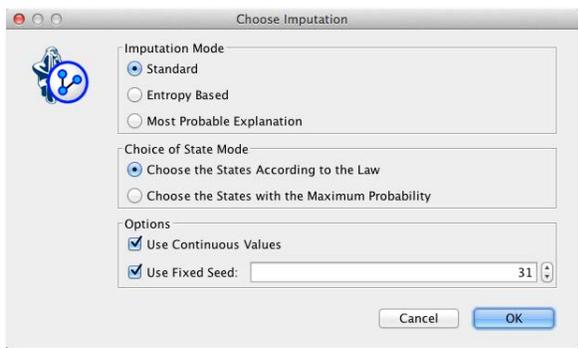
matically as we go along.⁸ However, there is a significant computational burden associated with this ongoing computation.

In order to accelerate all subsequent tasks, we will fix the most recent imputation that was generated during the learning of the **Maximum Weight Spanning Tree**.

To perform this imputation, we first select all nodes, then right-click on any one of the nodes with missing values, and finally select **Imputation** from the **Contextual Menu**.

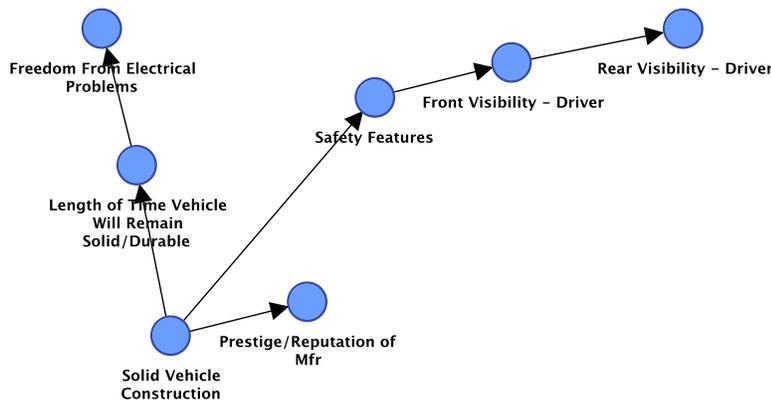


We are given a choice of modes, of which we select **Standard** and **Choose the States According to the Law**.



Upon completion of the process, all question marks disappear from the network, indicating that there are no more missing values.

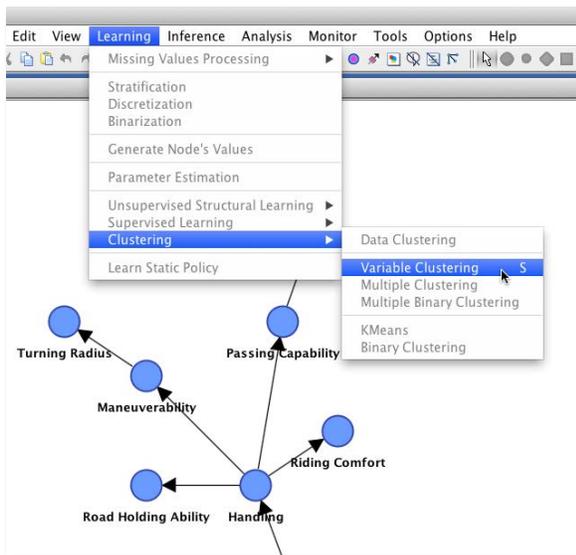
⁸ For more details, please see our white paper on missing values processing with Bayesian networks: <http://bayesia.us/missing-values-processing-with-bayesian-networks.html>



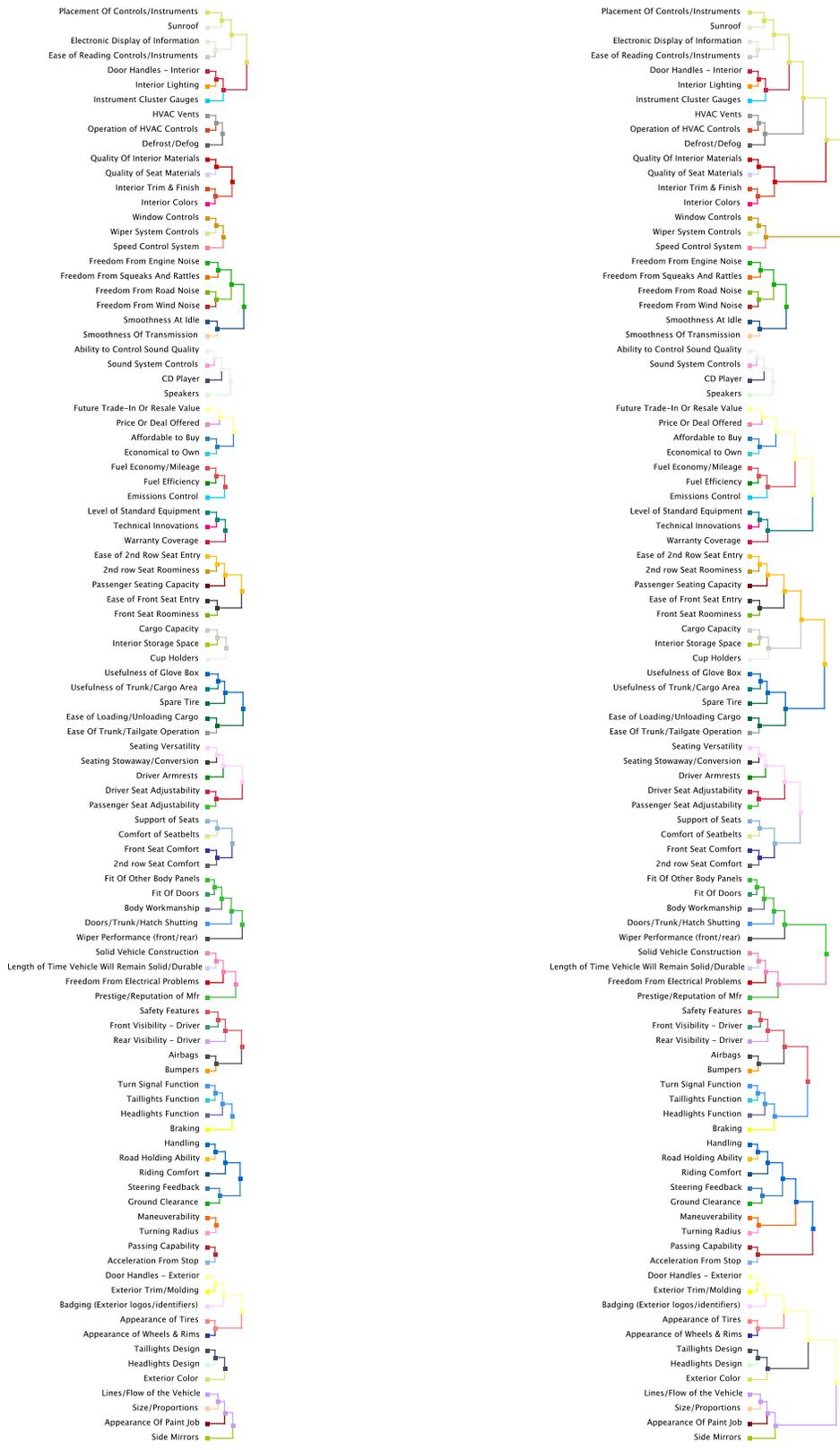
Variable Clustering

The network, as we see it here, is intended only as an interim step. For product planning or decision-making purposes, it would indeed be difficult to work directly with 98 manifest nodes. We would not be able to see the proverbial forest for all the trees. Rather, this network will serve as the basis for **Variable Clustering**, i.e. grouping nodes into meaningful concepts.

We start this clustering process, from within the **Validation Mode**, by selecting **Learning | Clustering | Variable Clustering** (or by using the keyboard shortcut “S”).

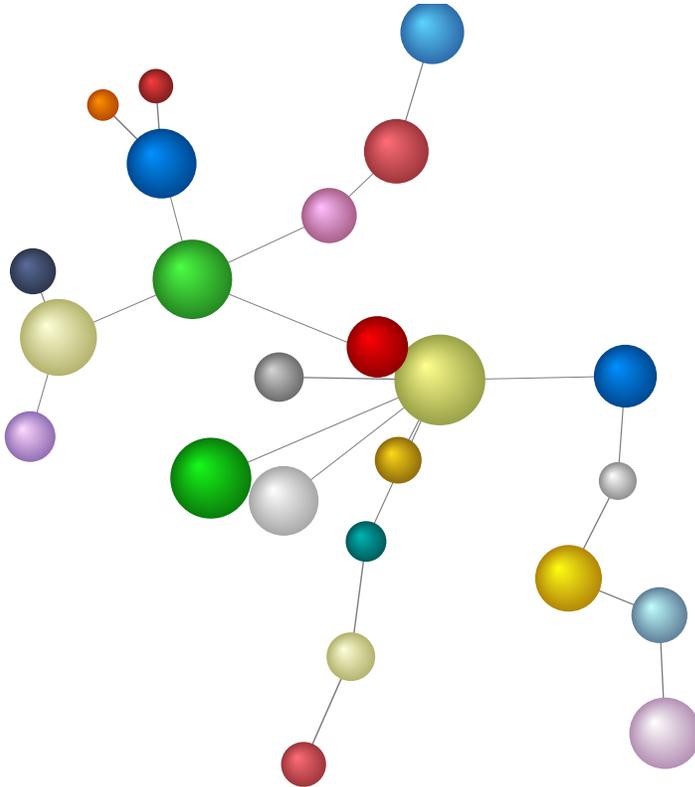


We now see the same graph as before; however, the nodes are now colored as per their proposed cluster membership. In our case, BayesiaLab suggests 26 clusters, as indicated in the menu bar.

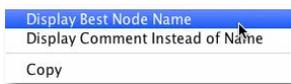


Alternatively, we can show the currently selected clustering in a similar way as in the previously-introduced

Mapping function. Click  in the menu bar to show this view:

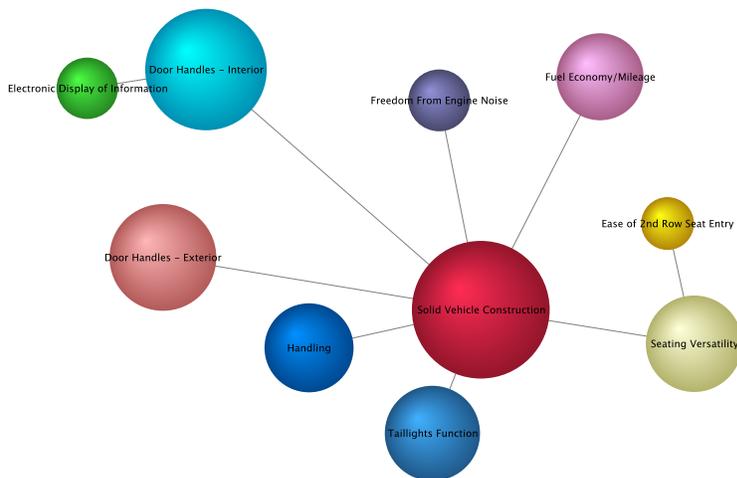
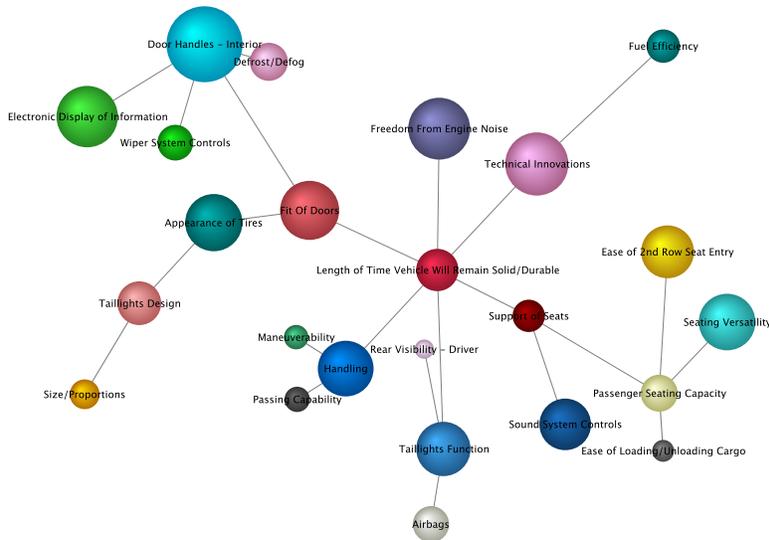


For an easier interpretation of these “bubbles”, we can attach labels to them. Since each bubble represents a cluster of nodes, we select **Display Best Node Name** from the **Contextual Menu**.

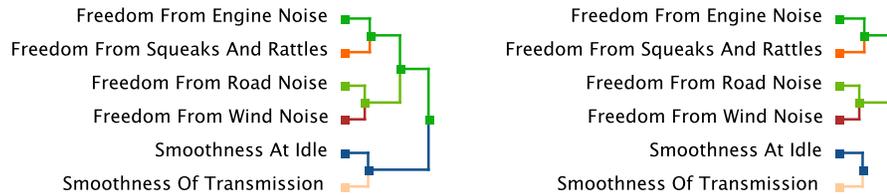


This shows the name of the node that most strongly contributes to each cluster, given the currently selected number of clusters.

Two screenshots are shown as examples, based on 24 and 10 clusters respectively.



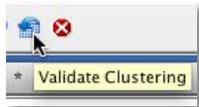
With **Dendrogram** and **Mapping**, we can visually experiment until the appropriate cluster number is established. The final selection of the number of clusters remains the task of the analyst. There is no hard-and-fast rule for choosing the number of clusters as this example illustrates. For instance, is it appropriate to cluster nodes related to *noise* with nodes related to *smoothness*? Two alternative **Dendrograms** are shown below. Only a domain expert can make a judgment in this regard.



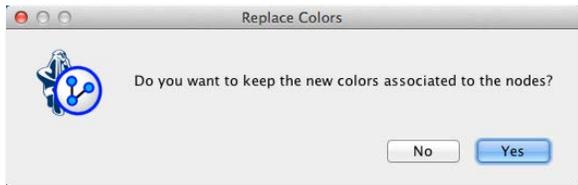
The number of clusters automatically proposed by BayesiaLab is based on two heuristics: the first is based on the strength of the relationships, the second on the maximum number of variables per cluster. Whereas we generally do not advise changing the former, the latter can be modified via **Options | Settings | Learning | Variable Clustering**.



After further review of all diagrams, we conclude that 24 clusters are most appropriate for this domain and confirm this choice by clicking the **Validate Clustering** button.



Furthermore, we confirm that we want to keep the colors from the just-completed interactive clustering.

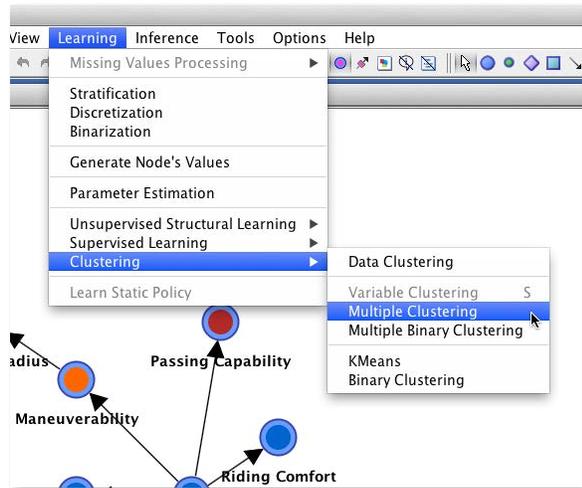


This confirmation generates the **Variable Clustering Report** that summarizes the nodes' cluster membership. Initially, the clusters are simply labeled as [Factor_0], [Factor_1], ..., [Factor_23].⁹

Latent Factor Induction via Multiple Clustering

As our next step, we introduce these newly-identified latent factors into our existing network and estimate their probabilistic relationships with the manifest variables. This means we create a new node for each latent factor, adding 24 new dimensions in our network. For this step, we need to return to the **Modeling Mode** because the introduction of the factor nodes into the networks requires the learning algorithms.

More specifically, we select **Learning | Clustering | Multiple Clustering**, which brings up the **Multiple Clustering** dialogue.

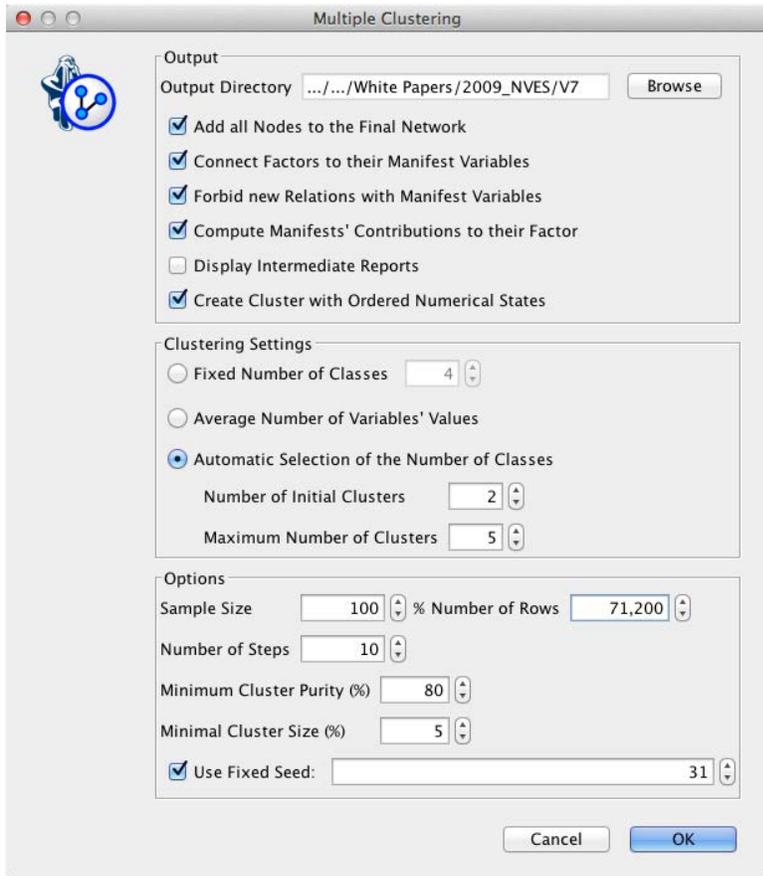


There is a range of settings, but we will focus only on a subset of the available options. Firstly, we need to specify an output directory for the to-be-learned subnetworks. Secondly, we need to set some parameters for the clustering process, such as the minimum and maximum number of states that can be created during the learning

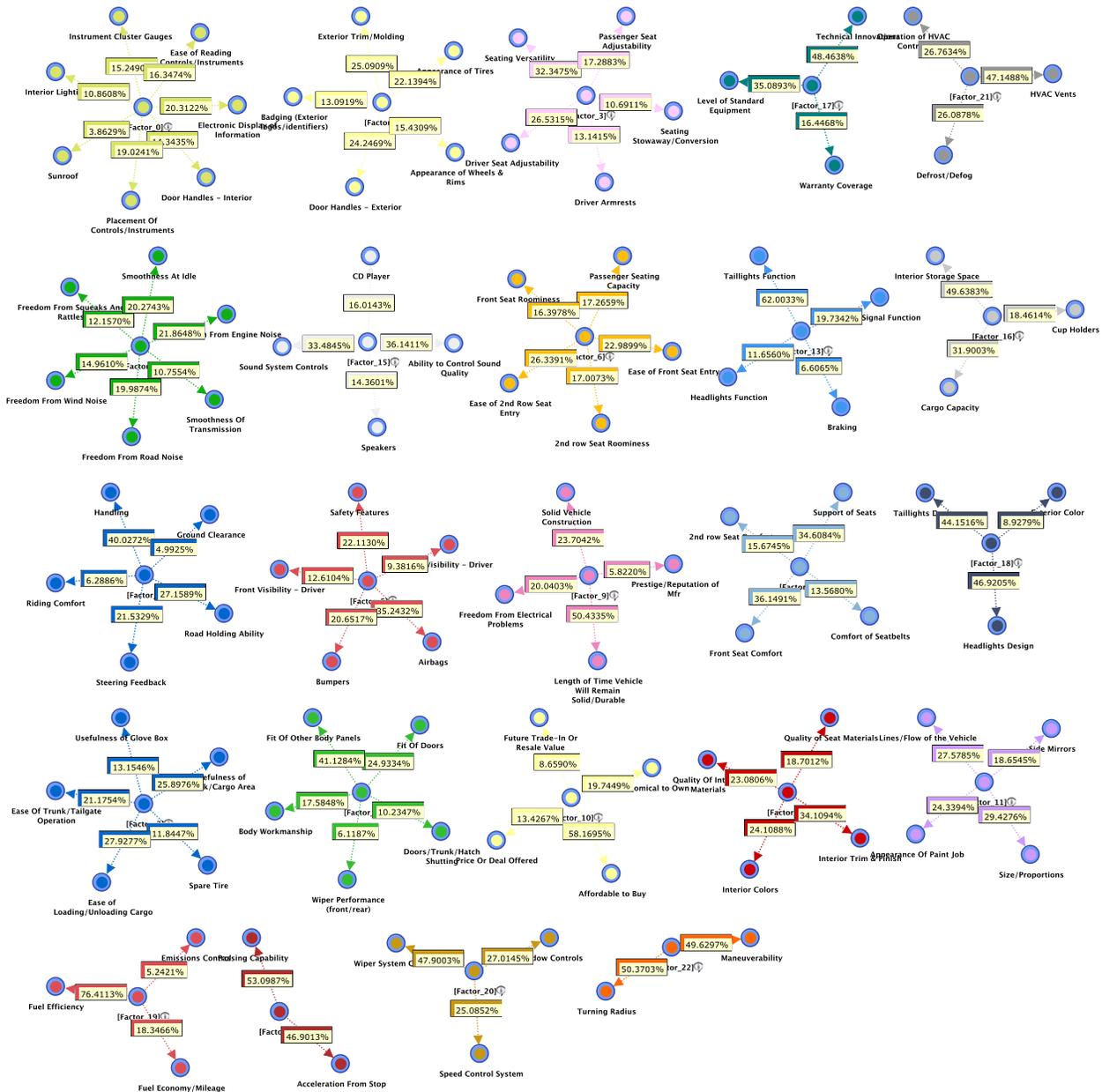
Classes	Nodes
[Factor_0]	Placement Of Controls/Instruments
	Sunroof
	Electronic Display of Information
	Instrument Cluster Gauges
	Interior Lighting
	Ease of Reading Controls/Instruments
	Door Handles - Interior
[Factor_1]	Smoothness Of Transmission
	Freedom From Squeaks And Rattles
	Freedom From Road Noise
	Smoothness At Idle
	Freedom From Wind Noise
[Factor_2]	Freedom From Engine Noise
	Usefulness of Trunk/Cargo Area
	Ease Of Trunk/Tailgate Operation
	Ease of Loading/Unloading Cargo
	Usefulness of Glove Box
[Factor_3]	Spare Tire
	Passenger Seat Adjustability
	Driver Seat Adjustability
	Driver Armrests
[Factor_4]	Seating Versatility
	Seating Stowaway/Conversion
	Doors/Trunk/Hatch Shutting
	Body Workmanship
[Factor_5]	Fit Of Doors
	Fit Of Other Body Panels
	Wiper Performance (front/rear)
	Safety Features
[Factor_6]	Front Visibility - Driver
	Rear Visibility - Driver
	Bumpers
[Factor_7]	Airbags
	Front Seat Roominess
	2nd row Seat Roominess
	Ease of Front Seat Entry
[Factor_8]	Ease of 2nd Row Seat Entry
	Passenger Seating Capacity
	Appearance of Tires
	Badging (Exterior logos/identifiers)
[Factor_9]	Door Handles - Exterior
	Appearance of Wheels & Rims
	Exterior Trim/Molding
	Ground Clearance
[Factor_10]	Riding Comfort
	Handling
	Road Holding Ability
	Steering Feedback
[Factor_11]	Length of Time Vehicle Will Remain Solid/Durable
	Freedom From Electrical Problems

⁹ A complete list of factors and their associated nodes is provided in the appendix.

process. For our example, we select **Automatic Selection of the Number of Classes**, which will allow the learning algorithm to find the optimum number of factor states up to a maximum of five states. This means that each new factor will need to represent the corresponding manifest variables with up to five states.



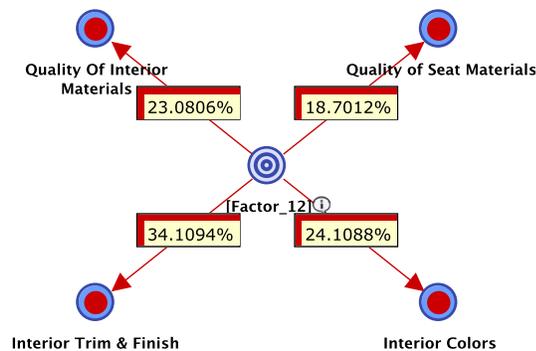
Upon completion of the **Multiple Clustering** process, we obtain a new network file that contains one small network for each cluster, with one factor being at the center of each cluster.



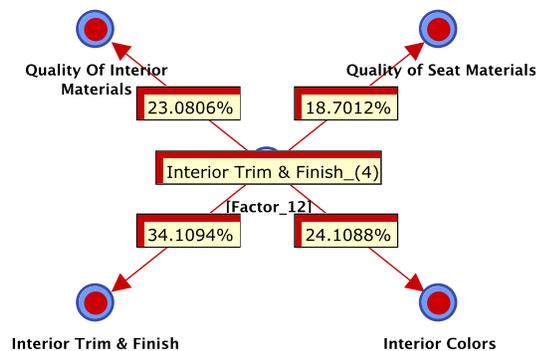
The arcs between the factors and their manifest nodes are labeled with **Direct Effect Contribution** values. This allows us to easily identify the importance of the manifests with regard to their respective factors.

Traditionally, we would now choose a name for each factor, so we can subsequently interpret the factors without looking at their manifests. For instance, *Factor_12*, shown below, could be called *Interior Quality* or something similar. In BayesiaLab, we have the option of deferring this naming process by simply using the

“strongest” node, as per the **Direct Effect Contribution** value, within each factor cluster as that factor’s **Node Comment**.



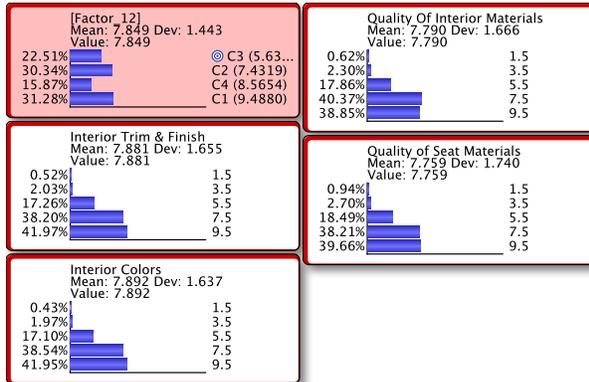
Clicking **Display Node Comments** in the menu bar will reveal *Interior Trim & Finish_(4)* as a label on the factor. The suffix “(4)” indicates that 4 manifest variables are linked to this factor.



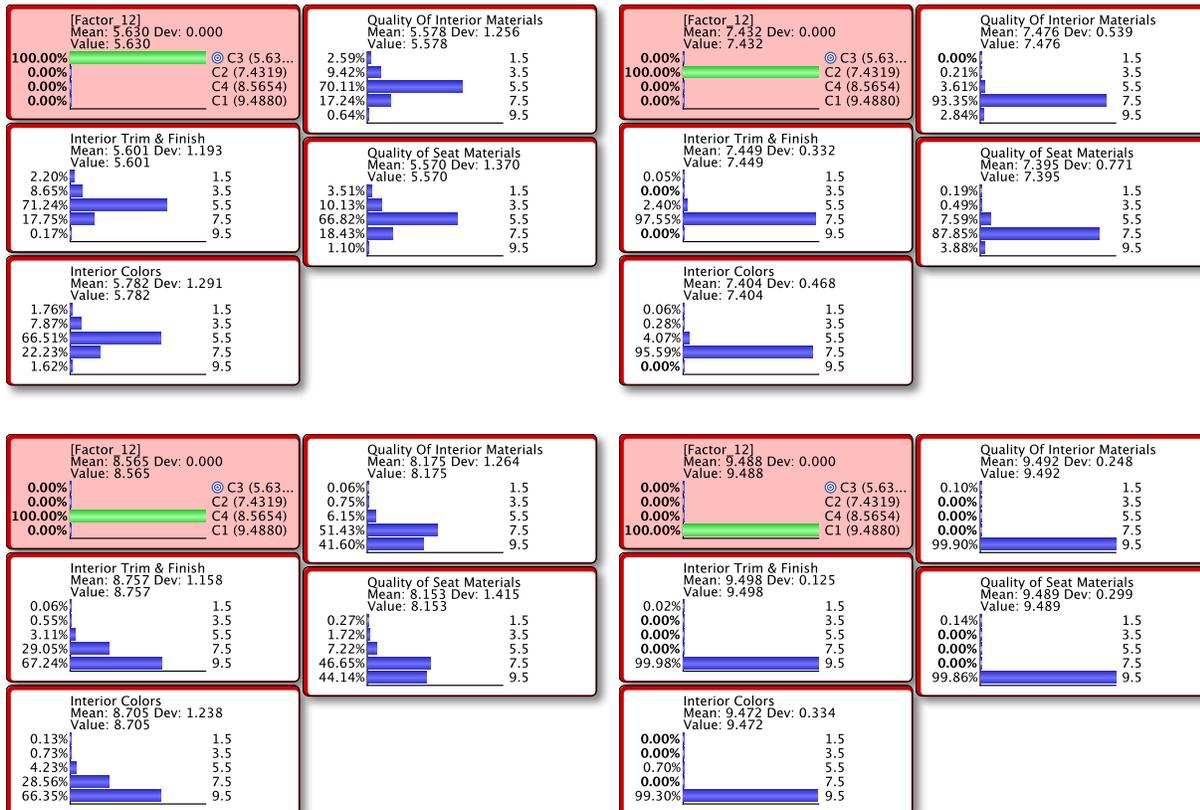
Factor States/Values

Beyond adding the factors to the network, the **Multiple Clustering** process has also generated states for all factors and computed their values. Inducing a factor means finding an appropriate summary of the underlying joint probability distribution defined by the manifest nodes. In the previous example of *Factor_12*, this would mean that the states of *Factor_12* can summarize the following four nodes: *Interior Trim & Finish*, *Quality of Interior Materials*, *Interior Colors*, and *Quality of Seat Materials*.

We can examine the factor states and values by opening the network for *Factor_12*, switching into **Validation Mode**, and selecting all nodes for display in the **Monitor Panel**. By default, we see the marginal distributions of all the manifests and the factor.



By sequentially setting evidence on each of the four states of *Factor_12*, we see what states of the manifests correspond to the factor states.

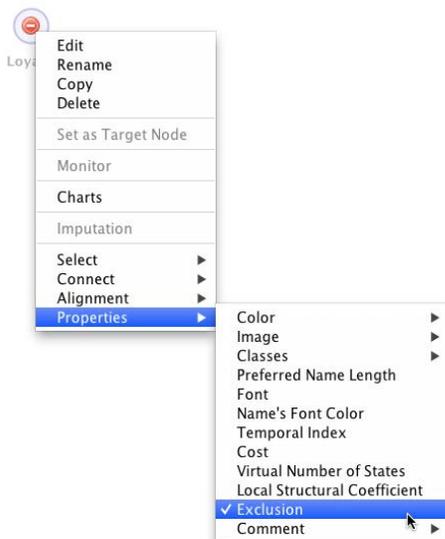


Looking at these **Monitors** also provides some intuition regarding the values of the states of *Factor_12*. BayesiaLab computes these values as the weighted average of the associated manifests' values. As such, *Factor_12* becomes a compact summary of the connected manifest nodes.

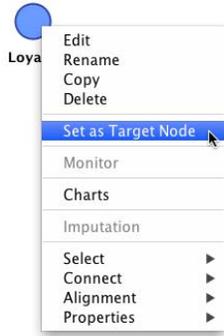
Introducing the Target Node

Now that factors have been formally introduced into the network, which each represent a major concept, we can proceed to the next step. We will introduce the principal variable of interest in this study, *Loyalty*, as the target variable.

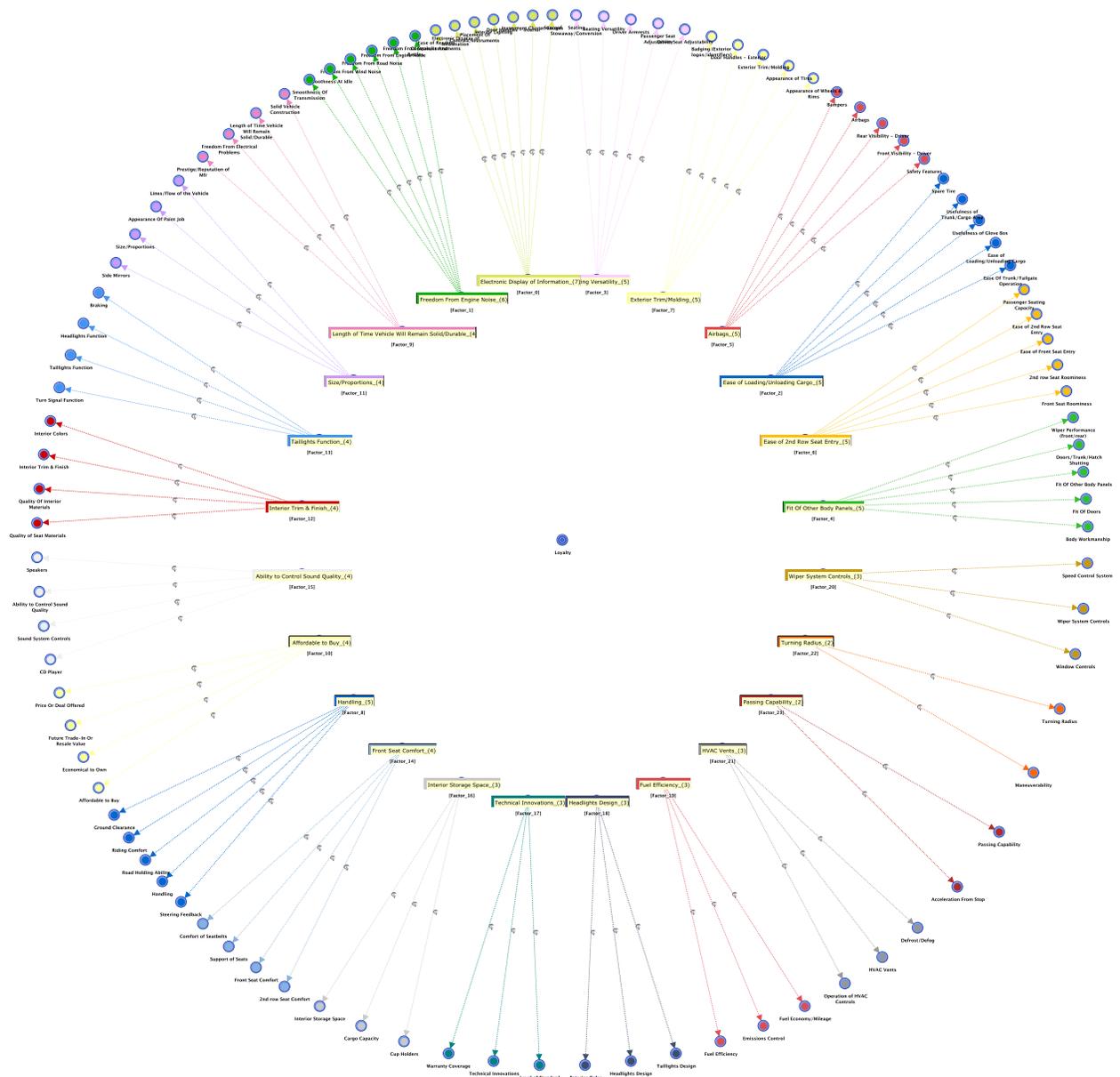
This node was excluded earlier in the clustering process, so it would not become clustered into a factor. So, the next step is to un-exclude this node, which we do by right-clicking the node and then selecting **Properties | Exclusion** (shortcut: press "X" and double-click on the node).



Also, we need to make this node the **Target Node**. We do this via by picking **Set as Target Node** from the contextual menu. Note that the un-exclusion and the **Target Node** definition can be done at the same time by pressing "T" and double-clicking on the excluded node.



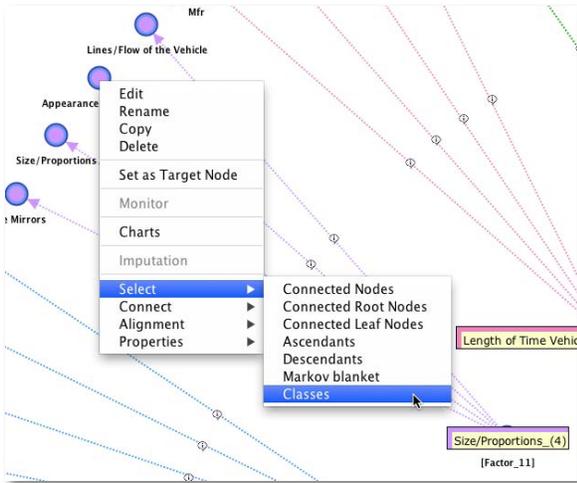
Upon introduction of the Target Node, we can interpret the status quo as the first two layers of a hierarchical model, as illustrated below. The outer ring contains the manifest nodes, the inner ring consists of the factors. In the middle, we have the yet-to-be-connected Target Node.



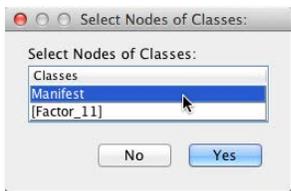
Focusing on Factors

We could continue our analysis with this network as is, including both factors and manifest nodes. However, for practical planning purposes, working with factors, i.e. the major concepts, is typically more relevant. Also, removing the manifest variable will improve the expository clarity of this tutorial. Thus, we will conduct all subsequent analyses exclusively with the factors, rather than the manifest nodes.

To delete the manifest nodes, we right-click on any one of them and then choose **Select | Classes**.

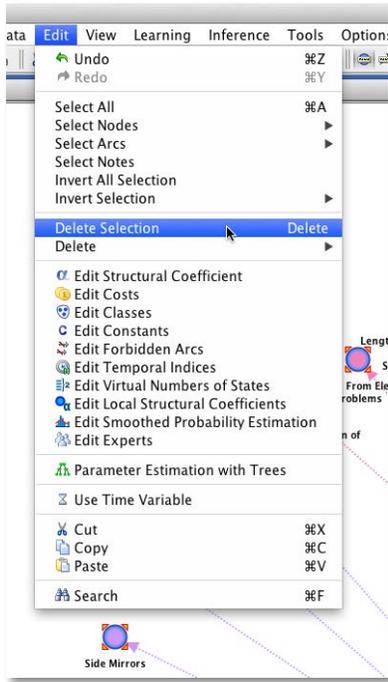


From the pop-up window, we pick **Manifest**.

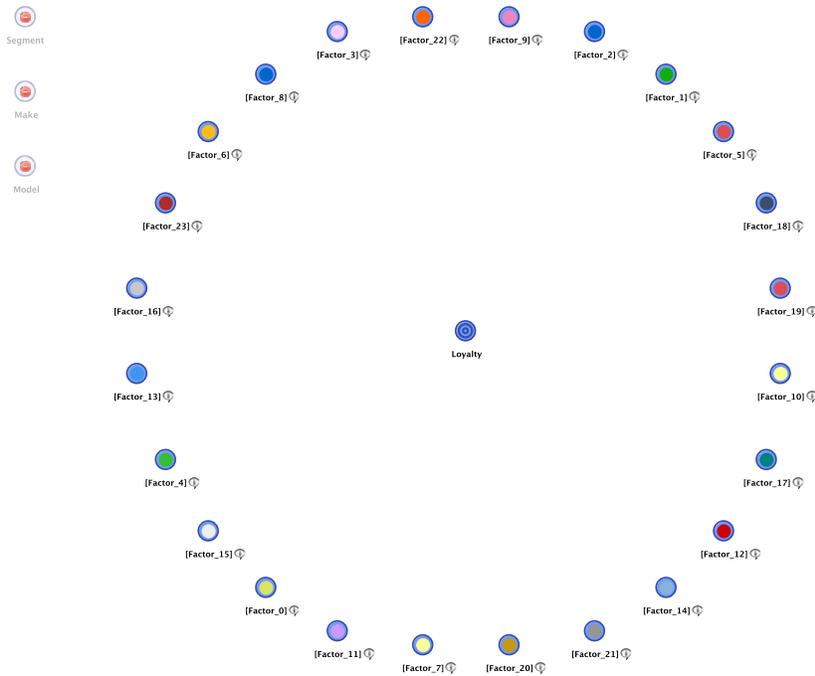


This highlights all manifest nodes, i.e. the outer ring. We can now delete them, either via the Delete key or from the main menu via **Edit | Delete Selection**.

Optimizing Customer Loyalty

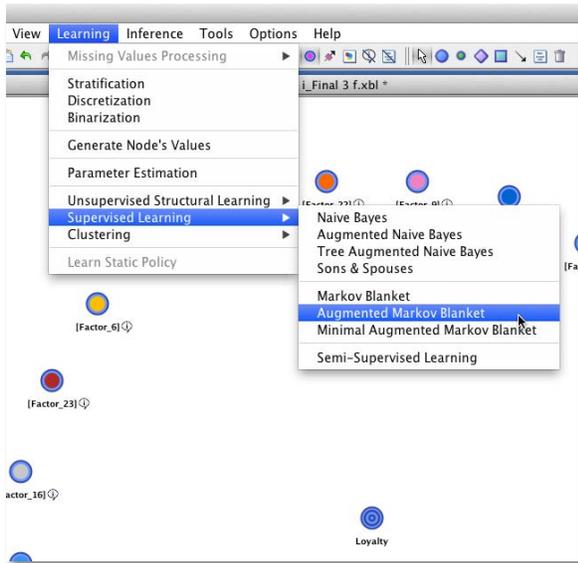


This leaves us with the factors, the **Target Node** *Loyalty*, plus the previously excluded nodes, *Segment*, *Make* and *Model*.

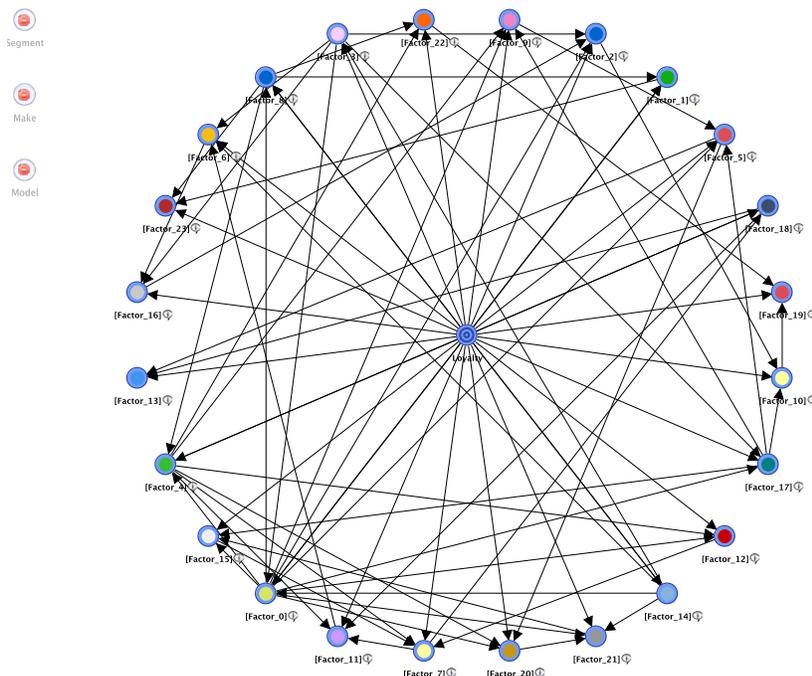


Supervised Learning

We can now use **Supervised Learning** to discover the relationships between the **Target Node** and the factors. We use the **Augmented Markov Blanket**, which is one of BayesiaLab's **Supervised Learning** algorithms.

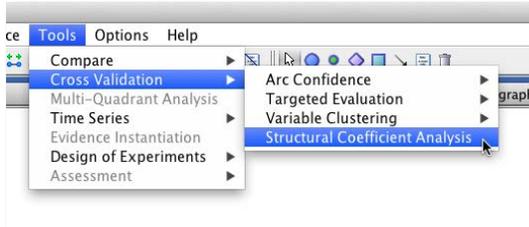


Performing this learning algorithm, using the default setting for the **Structural Coefficient** (SC=1), we obtain the following network:

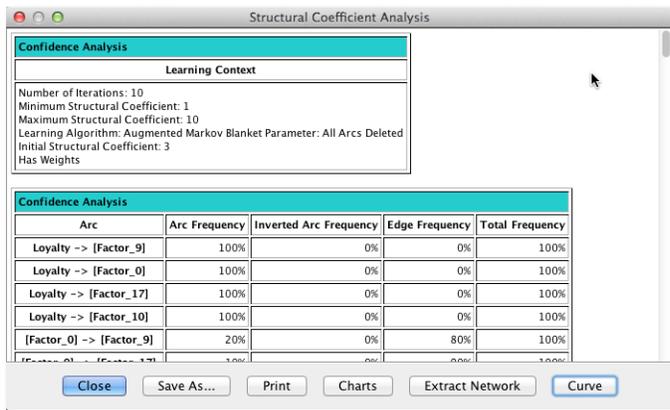


Structural Coefficient Analysis

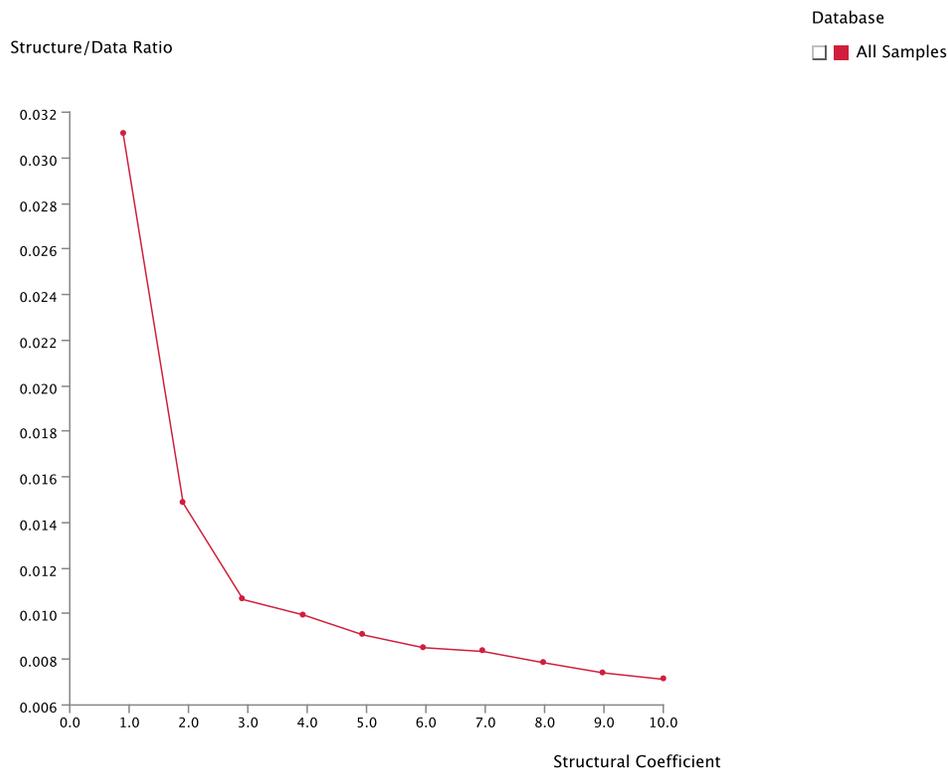
In the newly-learned network, we see a total of 88 arcs connecting the 24 factor nodes and the target. Some nodes have up to five parent nodes, which implies a 6-dimensional conditional probability table for those nodes. Given this rather high level of complexity of the network, it is prudent to perform a **Structural Coefficient Analysis: Tools | Cross Validation | Structural Coefficient Analysis**.



This way we can examine, among other metrics, the data-to-structure ratio as a function of the structural network complexity.

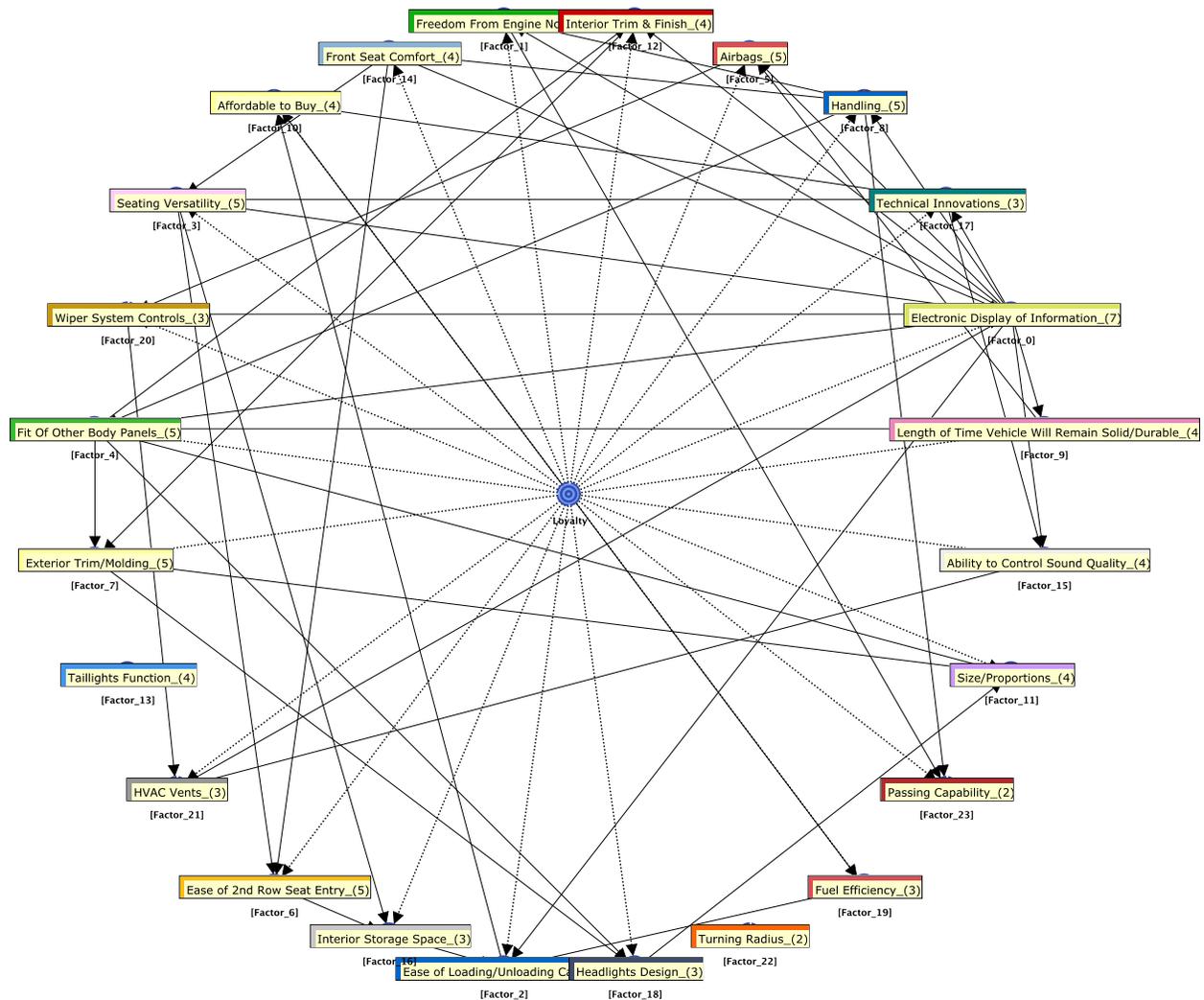


Once the report is presented, clicking **Curve** produces a kind of “scree plot”, which helps us identify a reasonable value of the **Structural Coefficient**. As opposed to the scree plot that we know from Factor Analysis, here, we read this plot from right to left.



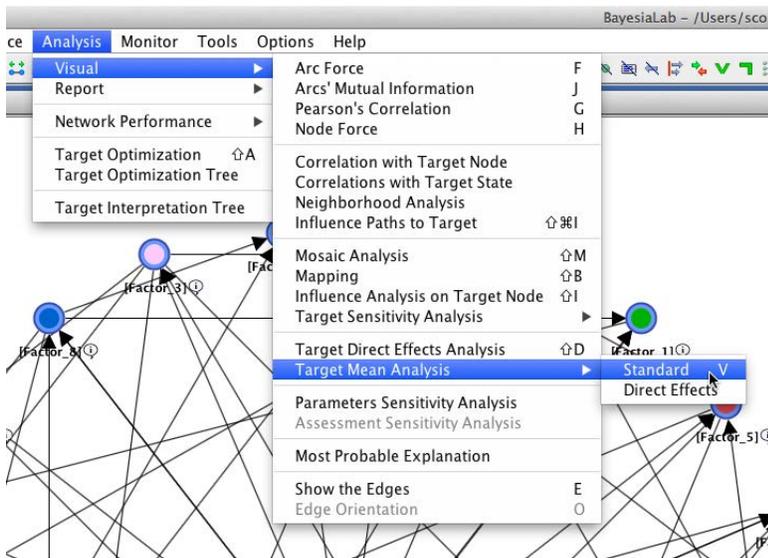
By visual inspection of this graph, moving from right to left along the x-axis, we see an inflection point of the curve around **SC=3**. Below that value, the structural complexity is increasing faster than the data likelihood. Thus, we choose **SC=3** and relearn the network on that basis with the **Augmented Markov Blanket** algorithm.

The resulting network is quite a bit simpler than before, now featuring only 65 arcs. Also, the *Turning Radius* and *Tailights Function* factors are now no longer part of the network, which suggests that these two factors are least relevant with regard to loyalty.

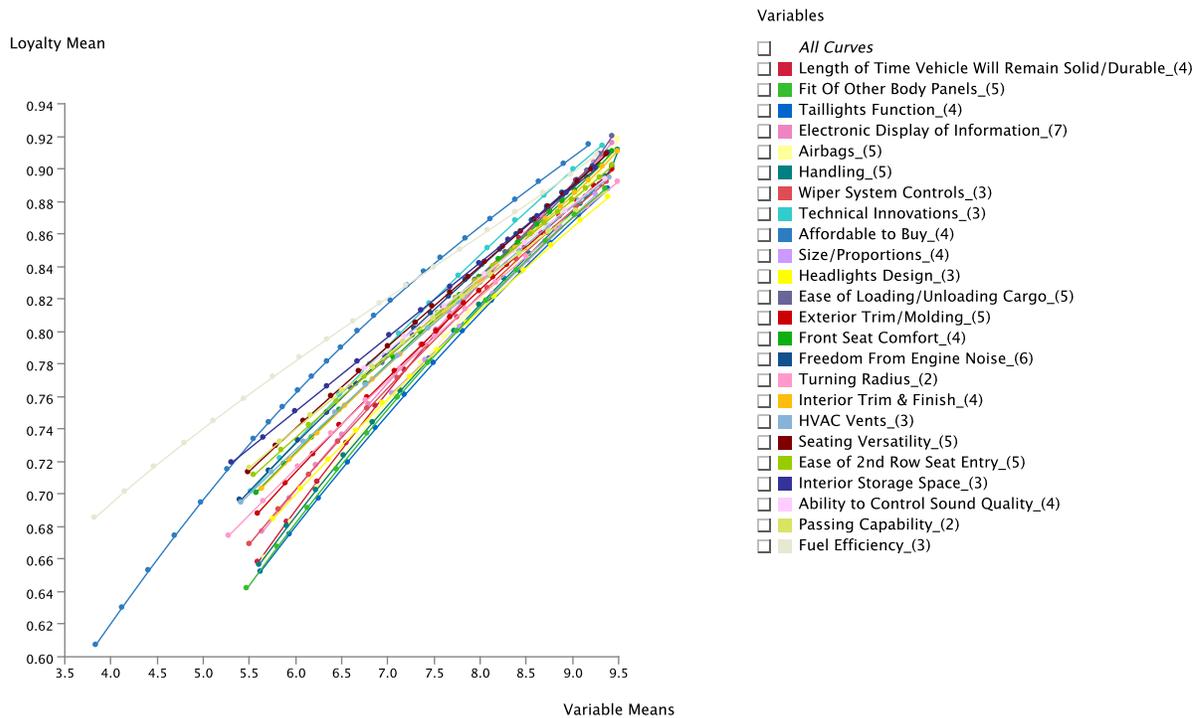


Target Mean Analysis

On the basis of this network structure, we can now examine the relationships between the factors and the target node. For this step, we select **Analysis | Visual | Target Mean Analysis | Standard**. This function computes the mean value of the **Target Node** by varying each factor, one at a time, across its entire range of values.

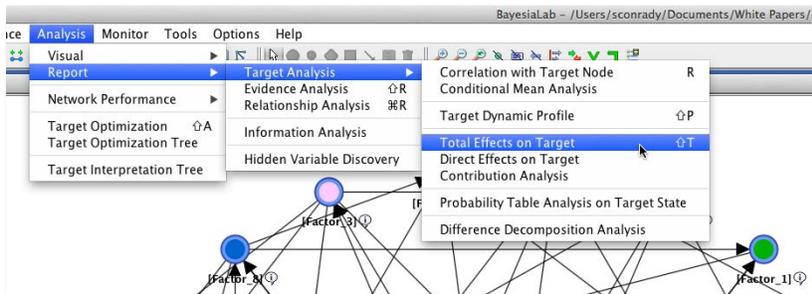


The **Target Mean Analysis** provides a quick overview of how our factors values are associated with the **Target Node**. The y-axis shows the mean values of the **Target Node** as a function of the factor values on the x-axis.



This plot suggests that all the factors are approximately linearly associated with the **Target Node**. Furthermore, the curves appear to run almost parallel between the x-values of 7.5 and 9. As a result, it is reasonable to formally compute “parameter estimates” for the slopes of these curves.

In BayesiaLab, this can be done by means of simulation via **Analysis | Reports | Target Analysis | Total Effects on Target**. More specifically, BayesiaLab computes the derivative around the mean value of the x-range of each factor.



The results are presented in a table. The **Total Effects** column shows the change of the mean value of the **Target Node**, given the observation of a one-unit change in each of the factors. This value is what we commonly interpret as slope.

Total Effects on Target (Associated graph 1b i_Final 3 f2)

Analysis Context

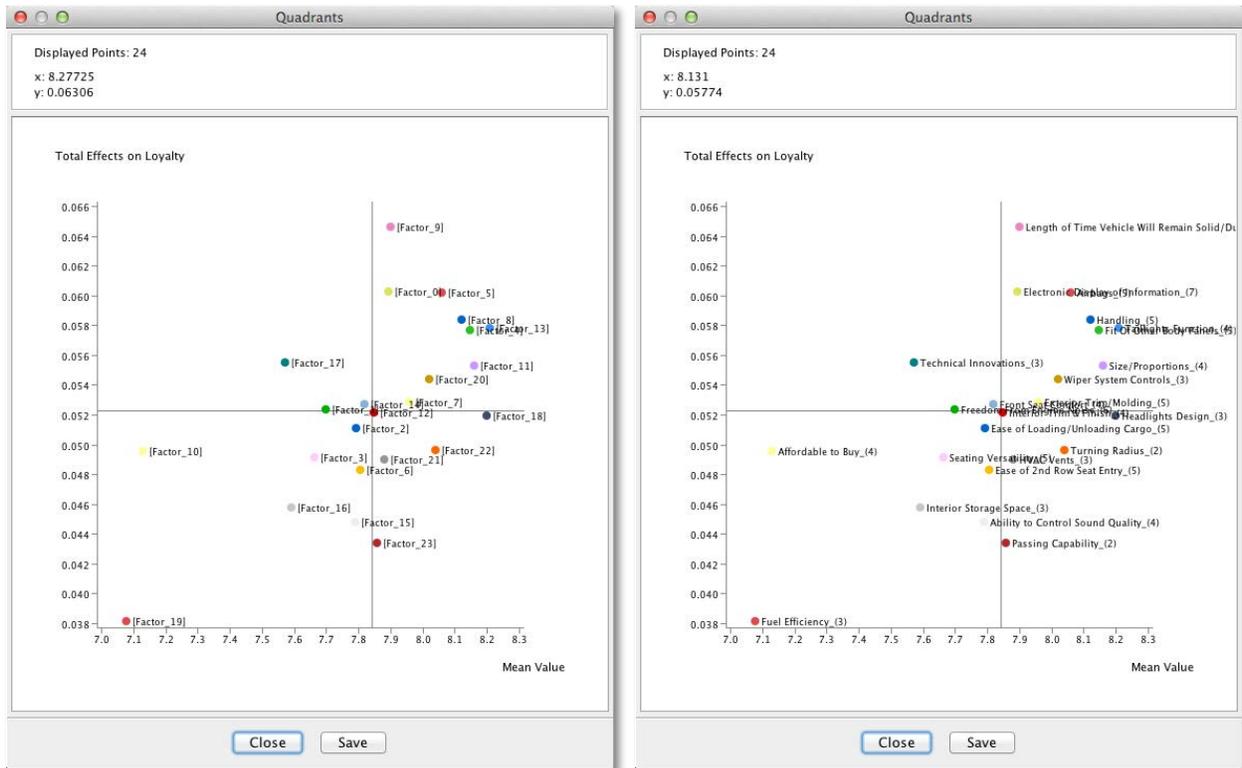
No Observation

Total Effects on Target Loyalty

Node	Value/Mean	Standardized Total Effects	Total Effects	G-test	Degrees of Freedom	p-value	G-test (Data)	Degrees of Freedom (Data)	p-value (Data)
[Factor 9]	7.9013	0.4245	0.0646	14,908.7833	8	0.0000%	14,782.5586	8	0.0000%
[Factor 0]	7.8951	0.3884	0.0603	13,282.9575	12	0.0000%	13,282.8311	12	0.0000%
[Factor 17]	7.5734	0.3866	0.0555	12,694.0856	8	0.0000%	12,693.9307	8	0.0000%
[Factor 8]	8.1228	0.3750	0.0584	12,446.1396	12	0.0000%	12,446.0303	12	0.0000%
[Factor 10]	7.1330	0.3701	0.0496	11,561.3423	16	0.0000%	11,682.0439	16	0.0000%
[Factor 5]	8.0608	0.3687	0.0602	12,062.7355	12	0.0000%	12,040.7812	12	0.0000%
[Factor 1]	7.6995	0.3658	0.0524	11,208.2514	12	0.0000%	11,208.1729	12	0.0000%
[Factor 12]	7.8492	0.3637	0.0522	11,381.3565	12	0.0000%	11,381.2529	12	0.0000%
[Factor 4]	8.1484	0.3602	0.0577	10,734.1634	8	0.0000%	10,734.5195	8	0.0000%
[Factor 14]	7.8198	0.3579	0.0527	11,083.0511	12	0.0000%	11,082.8965	12	0.0000%
[Factor 20]	8.0229	0.3566	0.0544	10,751.5944	8	0.0000%	10,740.2012	8	0.0000%
[Factor 3]	7.6649	0.3523	0.0492	10,823.0607	12	0.0000%	10,823.0459	12	0.0000%
[Factor 13]	8.2115	0.3466	0.0578	9,989.0221	8	0.0000%	9,847.1016	8	0.0000%
[Factor 7]	7.9603	0.3450	0.0529	10,322.4369	12	0.0000%	10,286.5332	12	0.0000%
[Factor 21]	7.8838	0.3419	0.0490	9,718.4888	8	0.0000%	9,827.9238	8	0.0000%
[Factor 6]	7.8065	0.3354	0.0484	9,858.3638	12	0.0000%	9,843.4473	12	0.0000%
[Factor 22]	8.0431	0.3318	0.0496	8,947.1285	8	0.0000%	8,947.1113	8	0.0000%
[Factor 16]	7.5940	0.3299	0.0458	9,318.8270	8	0.0000%	9,309.5576	8	0.0000%
[Factor 18]	8.2025	0.3243	0.0520	8,974.5122	8	0.0000%	8,964.0029	8	0.0000%
[Factor 19]	7.0787	0.3240	0.0382	8,817.0779	12	0.0000%	8,885.0605	12	0.0000%
[Factor 23]	7.8610	0.3180	0.0434	8,390.6899	8	0.0000%	8,390.8027	8	0.0000%
[Factor 2]	7.7942	0.3056	0.0511	9,131.3577	8	0.0000%	8,882.7549	8	0.0000%
[Factor 11]	8.1607	0.2945	0.0553	7,634.3680	4	0.0000%	7,601.8452	4	0.0000%
[Factor 15]	7.7920	0.2768	0.0448	7,189.1224	4	0.0000%	7,250.4067	4	0.0000%

Close Save As... Print Quadrants

Clicking **Quadrants** on the report window shows a scatterplot with factor values on the x-axis and **Total Effects** on the y-axis. This allows us to better distinguish the factors, even though they have **Total Effects** in a fairly narrow range around 0.4 to 0.6.

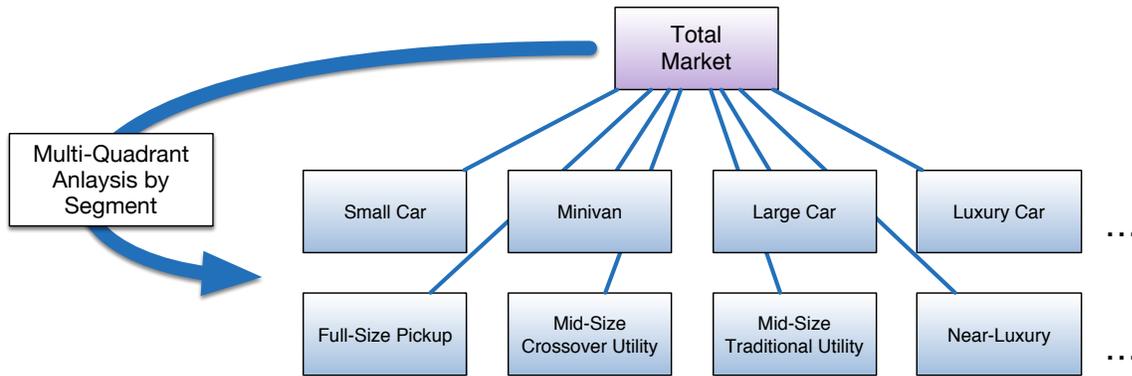


With the highest **Total Effect**, *Length of Time Vehicle Will Remain Solid/Durable* marks the top value on the y-axis of the plot. The factor *Fuel Efficiency* marks the bottom end on both axes. The position of the *Fuel Efficiency* factor is perhaps curious as our survey data covers 2009, when the auto industry was most severely affected by the recession.

However, as interesting as this may seem, it is probably little practical use for planning purposes as this plot represents a view of the entire market, across all makes and all segments. It is reasonable to assume that effect heterogeneity exists between vehicle segments as different a *Full-Size Pickups* and *Luxury Sedans*.

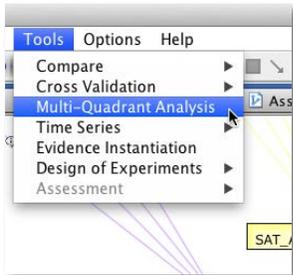
Multi-Quadrant Analysis (Total Market → Segment)

To study this domain at the level of vehicle segments, we could now start all over again and generate a new network for each segment from scratch. BayesiaLab provides a convenient shortcut for the researcher by means of **Multi-Quadrant Analysis**.



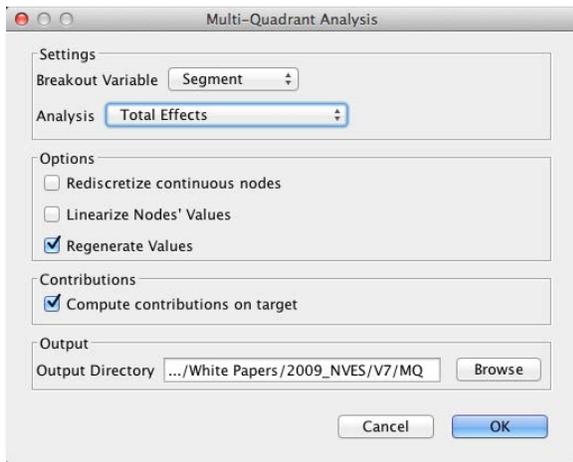
We can have BayesiaLab automatically replicate the original model (learned for the entire market) for each state of the to-be-specified **Breakout Node**. This is when the previously excluded node, *Segment* come into play. To make use of it here, we need to un-exclude it at this time.

We start the **Multi-Quadrant Analysis** from the main menu, within **Validation Mode**, via **Tools | Multi-Quadrant Analysis**.



In the following dialog box, we specify the options of the **Multi-Quadrant Analysis**. Most importantly, we need to select the **Breakout Variable**, which in our case needs to be *Segment*. Furthermore, we define an output directory. This is where the segment-level networks will be saved.

Optimizing Customer Loyalty

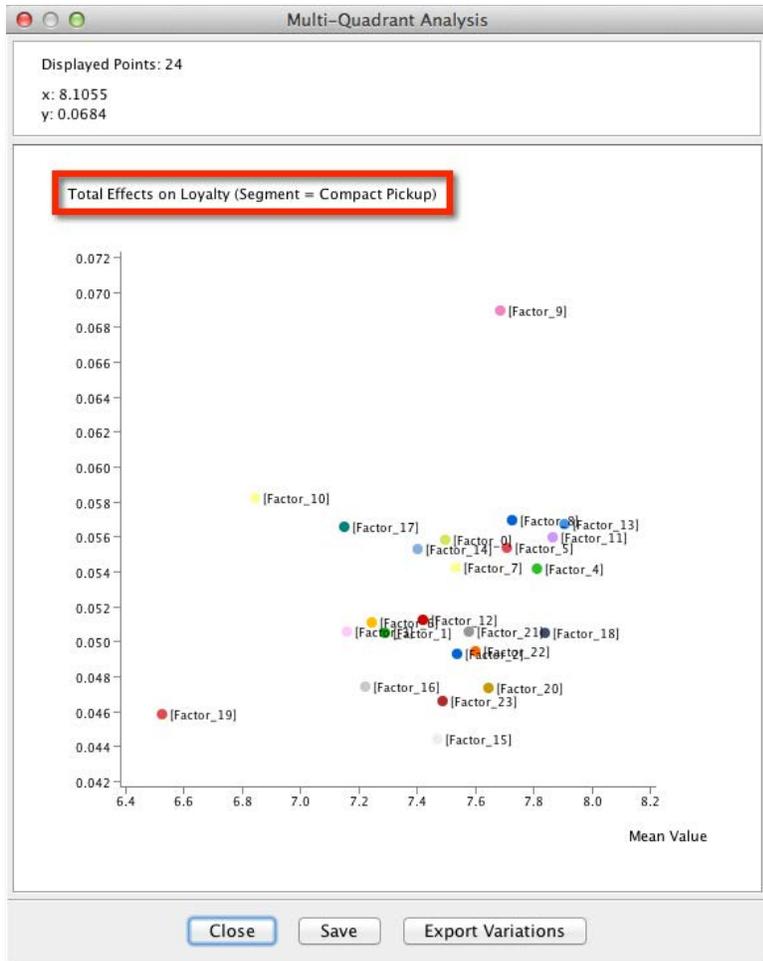


Once this process is completed, all new networks can be found in the specified directory. The file names are created according the following syntax: *Original Network File Name + _MULTI_QUADRANT_ + Breakout Variable State*.

Name	Date Modified	Size	Kind
NVES_MULTI_QUADRANT_Compact Pickup.xbl	Today, 9:26 PM	890 KB	BayesiaLab Network File
NVES_MULTI_QUADRANT_Convertible.xbl	Today, 9:26 PM	755 KB	BayesiaLab Network File
NVES_MULTI_QUADRANT_Eco-Car.xbl	Today, 9:26 PM	801 KB	BayesiaLab Network File
NVES_MULTI_QUADRANT_Entry Utility.xbl	Today, 9:26 PM	1.1 MB	BayesiaLab Network File
NVES_MULTI_QUADRANT_Full Size Cargo Van.xbl	Today, 9:26 PM	439 KB	BayesiaLab Network File
NVES_MULTI_QUADRANT_Full-Size Passenger Van.xbl	Today, 9:26 PM	399 KB	BayesiaLab Network File
NVES_MULTI_QUADRANT_Full-Size Pickup.xbl	Today, 9:26 PM	843 KB	BayesiaLab Network File
NVES_MULTI_QUADRANT_Heavy Duty Pickup.xbl	Today, 9:26 PM	672 KB	BayesiaLab Network File
NVES_MULTI_QUADRANT_Large Car.xbl	Today, 9:26 PM	900 KB	BayesiaLab Network File
NVES_MULTI_QUADRANT_Large Utility.xbl	Today, 9:26 PM	725 KB	BayesiaLab Network File
NVES_MULTI_QUADRANT_Luxury Car.xbl	Today, 9:26 PM	864 KB	BayesiaLab Network File
NVES_MULTI_QUADRANT_Luxury Multi-Function.xbl	Today, 9:26 PM	632 KB	BayesiaLab Network File
NVES_MULTI_QUADRANT_Luxury Pickup.xbl	Today, 9:26 PM	468 KB	BayesiaLab Network File
NVES_MULTI_QUADRANT_Luxury Utility.xbl	Today, 9:26 PM	812 KB	BayesiaLab Network File
NVES_MULTI_QUADRANT_Mid-Size Car.xbl	Today, 9:26 PM	1.1 MB	BayesiaLab Network File
NVES_MULTI_QUADRANT_Mid-Size Crossover Utility.xbl	Today, 9:26 PM	984 KB	BayesiaLab Network File
NVES_MULTI_QUADRANT_Mid-Size Multi-Function.xbl	Today, 9:26 PM	790 KB	BayesiaLab Network File
NVES_MULTI_QUADRANT_Mid-Size Traditional Utility.xbl	Today, 9:26 PM	793 KB	BayesiaLab Network File
NVES_MULTI_QUADRANT_Mid-Specialty.xbl	Today, 9:26 PM	680 KB	BayesiaLab Network File
NVES_MULTI_QUADRANT_Minivan.xbl	Today, 9:26 PM	905 KB	BayesiaLab Network File
NVES_MULTI_QUADRANT_Near Luxury Utility.xbl	Today, 9:26 PM	942 KB	BayesiaLab Network File
NVES_MULTI_QUADRANT_Near Luxury.xbl	Today, 9:26 PM	1 MB	BayesiaLab Network File
NVES_MULTI_QUADRANT_Premium Convertible Roadster.xbl	Today, 9:27 PM	739 KB	BayesiaLab Network File
NVES_MULTI_QUADRANT_Premium Coupe.xbl	Today, 9:27 PM	791 KB	BayesiaLab Network File
NVES_MULTI_QUADRANT_Small Car.xbl	Today, 9:27 PM	975 KB	BayesiaLab Network File
NVES_MULTI_QUADRANT_Small Multi-Function.xbl	Today, 9:27 PM	1 MB	BayesiaLab Network File
NVES_MULTI_QUADRANT_Specialty Coupe.xbl	Today, 9:27 PM	918 KB	BayesiaLab Network File

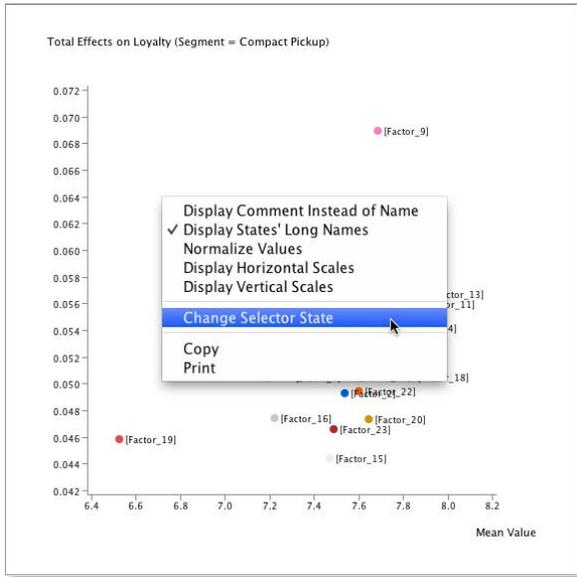
In BayesiaLab itself, we obtain a **Quadrant Plot**, which shows the **Mean Value** of each node on the x-axis and the **Total Effect** on the y-axis (even though quadrants are not explicitly shown here, we will soon explain how a quadrant view can be helpful for interpretation).

This plot exists for all the states of the **Breakout Variable**, i.e. for all segments. The currently selected state is highlighted.



By default, the **Breakout Variable's** first state is shown, in our case, *Compact Pickup*. To see the results of other segments, we right-click on the plot and pick **Change Selector State**.

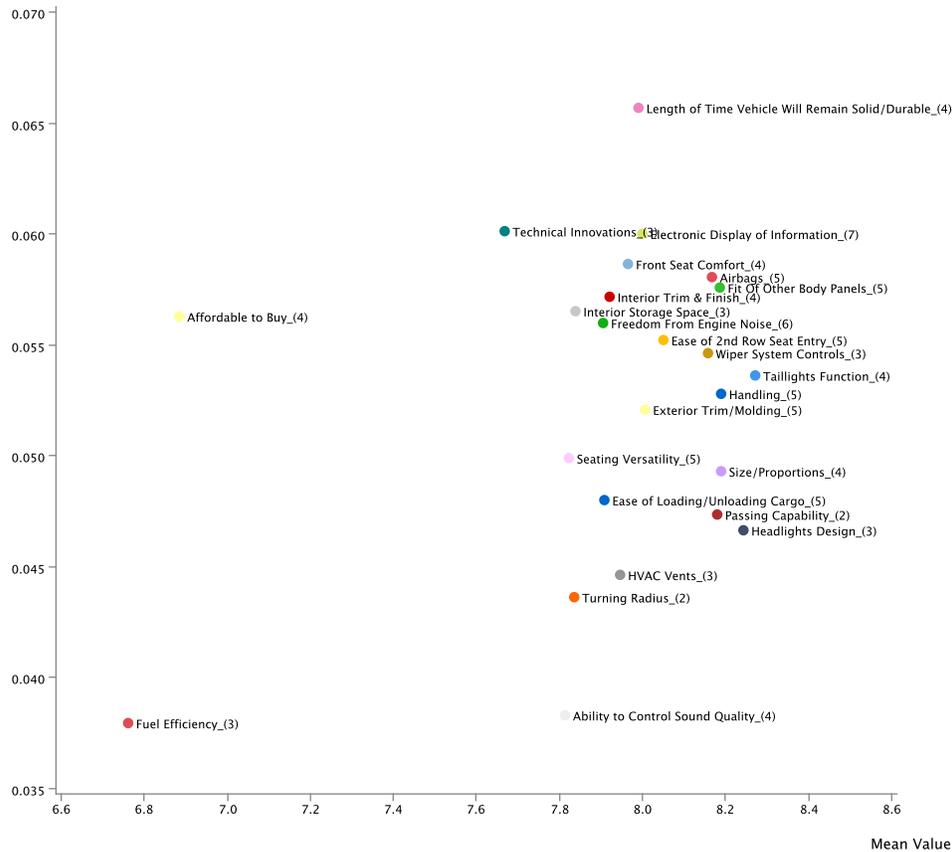
Optimizing Customer Loyalty



Full-Size Pickup Segment

For reasons explained in the introduction, we will now focus on the *Full-Size Pickup* segment.

Total Effects on Loyalty (Segment = Full-Size Pickup)

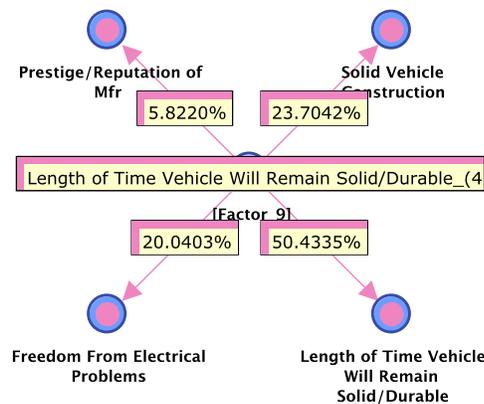


This plot allows immediate interpretation. The x-axis can be represents the mean satisfaction of *Full-Size Pickup* buyers with regard to the factors. The y-axis shows the **Total Effect** of each factor with regard to *Loyalty*. More specifically, the y-axis shows the value associated with a one-unit change of respective factor. Casually speaking, we interpret this as the “importance” of a variable. For *Full-Size Pickup*, this would mean that *Ability to Control Sound Quality* is fairly unimportant for *Loyalty*. On the other hand, even though *Affordable to Buy* rates low on the x-axis, it rates fairly high on the y-axis. This means that it is rather important for *Loyalty*.

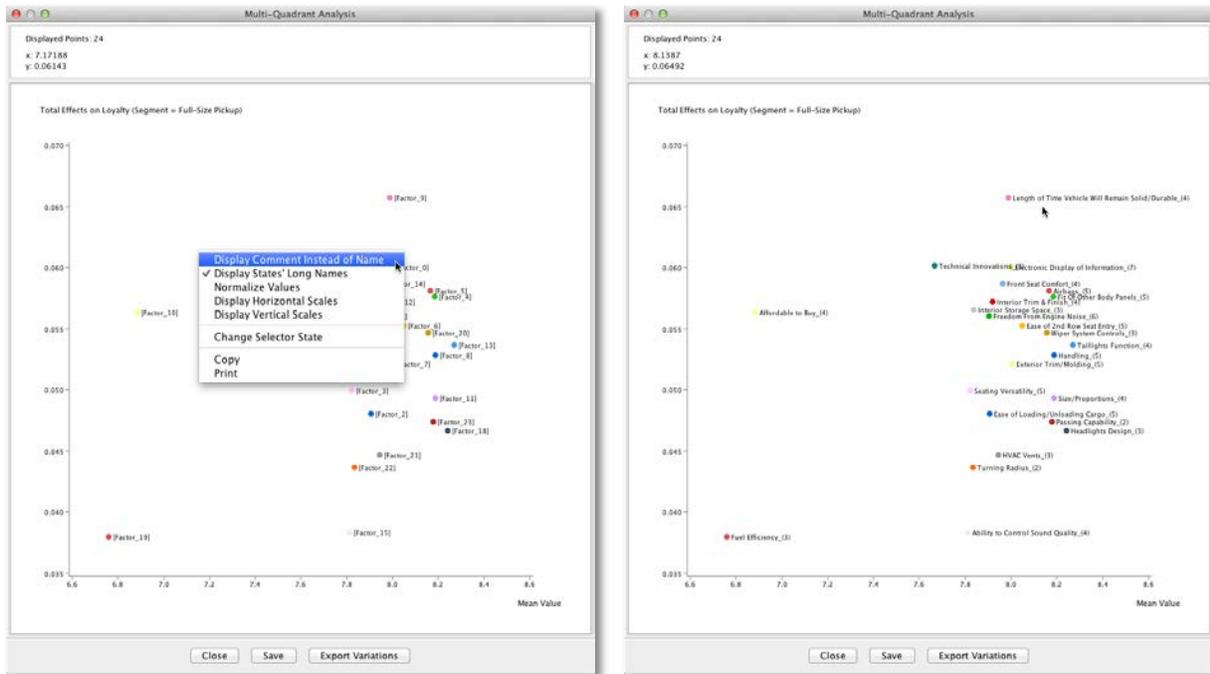
The following conceptual diagram shows a commonly-used interpretation



We should emphasize that we are interpreting factors, rather than manifest variables. Thus, the apparent “top driver” in the *Full-Size Pickup* plot, *Length of Time Vehicle Will Remain Solid/Durable*, is actually *Factor_9*. Only for convenience we applied the name of the node that most strongly contributes to this factor as node comment. For reference, the manifest nodes associated with this factor are shown below.



In the **Quadrant Plot**, we can easily toggle between the factor name, i.e. *[Factor_x]*, and the **Node Comment** via the contextual menu.



All the factors' positions on the **Quadrant Plot** become even more meaningful in the context of other segments. Within the same **Quadrant Plot** window, we can hover over any of the factors to see how other segments compare on the selected attribute.

The following screenshot shows the positions of all segments with regard to *Factor 17*, which is labeled *Length of Time Vehicle Will Remain Solid/Durable* .

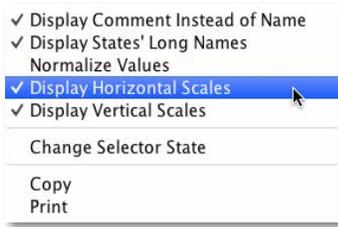
Optimizing Customer Loyalty

Total Effects on Loyalty (Segment = Full-Size Pickup)



This plot would suggest, for instance, that the *Full Size Cargo Van* segment has opportunities in this context. The *Premium Convertible/Roadster* segment, at the other end of the spectrum, might be in the “overkill” zone.

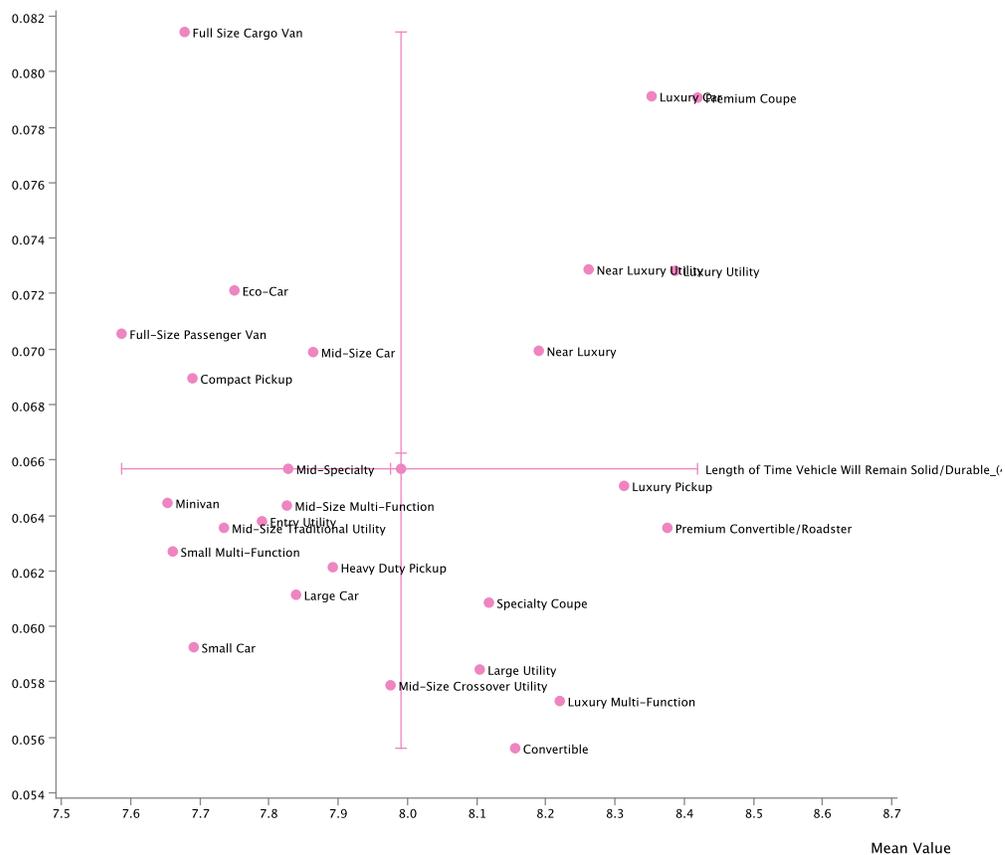
BayesiaLab offers a convenient way to see the relative position versus the segments. From the contextual menu, we can select **Display Horizontal/Vertical Scales**.



These scales show the range from the lowest to highest values. Additionally, a tick mark indicates the mean value of the respective attribute.

In the plot below, we show the **Total Effect** for *Length of Time Vehicle Will Remain Solid/Durable* for each segment. The intersection of the of the horizontal and vertical scale indicates the position of the *Full-Size Pickup* segment with regard to this variable.

Total Effects on Loyalty (Segment = Full-Size Pickup)



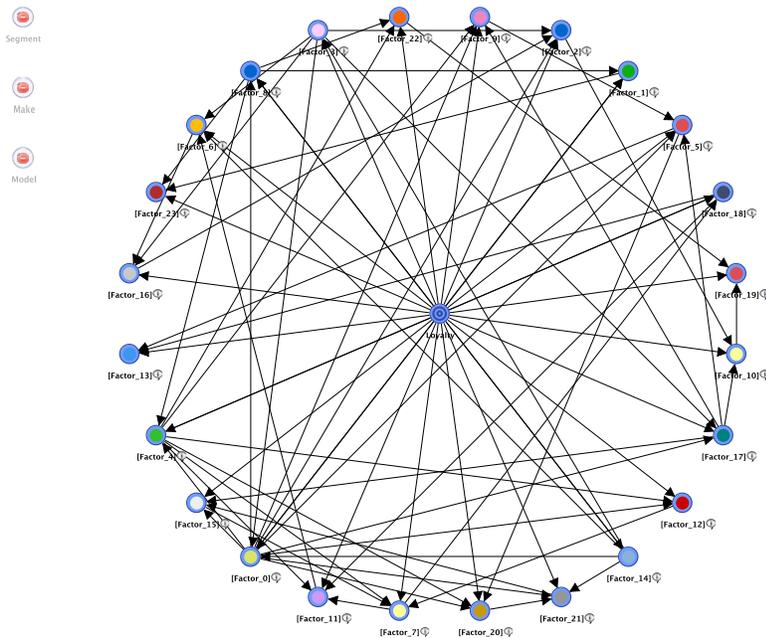
This analysis can certainly help us to understand the general areas that are important for loyalty in the individual segments. However, it does not provide any insight into the specific opportunities for individual vehicle models. For this, we need to proceed to the next level of detail, i.e. the model level.

Multi-Quadrant Analysis (Segment → Model)

During the earlier **Multi-Quadrant Analysis**, BayesiaLab generated one network file for each vehicle segment. We now open the network for the *Full-Size Pickup*, the focus of this study.

Name	Date...dified	Size	Kind
NVES_MC2_MULTI_QUADRANT_Mid-Size Car	12/22/13	389 KB	BayesiaLab Network File
NVES_MC2_MULTI_QUADRANT_Premium Coupe	12/22/13	210 KB	BayesiaLab Network File
NVES_MC2_MULTI_QUADRANT_Small Car	12/22/13	389 KB	BayesiaLab Network File
NVES_MC2_MULTI_QUADRANT_Small Multi-Function	12/22/13	378 KB	BayesiaLab Network File
NVES_MC2_MULTI_QUADRANT_Specialty Coupe	12/22/13	269 KB	BayesiaLab Network File
NVES_MC2_MULTI_QUADRANT_Near Luxury	12/22/13	443 KB	BayesiaLab Network File
NVES_MC2_MULTI_QUADRANT_Premium Convertible Roadster	12/22/13	185 KB	BayesiaLab Network File
NVES_MC2_MULTI_QUADRANT_Premium Coupe	12/22/13	210 KB	BayesiaLab Network File
NVES_MC2_MULTI_QUADRANT_Near Luxury Utility	12/22/13	291 KB	BayesiaLab Network File
NVES_MC2_MULTI_QUADRANT_Mid-Size Crossover Utility	12/22/13	379 KB	BayesiaLab Network File
NVES_MC2_MULTI_QUADRANT_Minivan	12/22/13	285 KB	BayesiaLab Network File
NVES_MC2_MULTI_QUADRANT_Mid-Size Traditional Utility	12/22/13	221 KB	BayesiaLab Network File
NVES_MC2_MULTI_QUADRANT_Mid-Specialty	12/22/13	168 KB	BayesiaLab Network File
NVES_MC2_MULTI_QUADRANT_Mid-Size Multi-Function	12/22/13	209 KB	BayesiaLab Network File
NVES_MC2_MULTI_QUADRANT_Luxury Utility	12/22/13	212 KB	BayesiaLab Network File
NVES_MC2_MULTI_QUADRANT_Entry Utility	12/22/13	543 KB	BayesiaLab Network File
NVES_MC2_MULTI_QUADRANT_Luxury Pickup	12/22/13	123 KB	BayesiaLab Network File
NVES_MC2_MULTI_QUADRANT_Large Car	12/22/13	252 KB	BayesiaLab Network File
NVES_MC2_MULTI_QUADRANT_Luxury Car	12/22/13	238 KB	BayesiaLab Network File
NVES_MC2_MULTI_QUADRANT_Large Utility	12/22/13	178 KB	BayesiaLab Network File
NVES_MC2_MULTI_QUADRANT_Luxury Multi-Function	12/22/13	154 KB	BayesiaLab Network File
NVES_MC2_MULTI_QUADRANT_Compact Pickup	12/22/13	266 KB	BayesiaLab Network File
NVES_MC2_MULTI_QUADRANT_Full Size Cargo Van	12/22/13	126 KB	BayesiaLab Network File
NVES_MC2_MULTI_QUADRANT_Full-Size Pickup	12/22/13	249 KB	BayesiaLab Network File
NVES_MC2_MULTI_QUADRANT_Full-Size Passenger Van	12/22/13	105 KB	BayesiaLab Network File
NVES_MC2_MULTI_QUADRANT_Heavy Duty Pickup	12/22/13	166 KB	BayesiaLab Network File
NVES_MC2_MULTI_QUADRANT_Convertible	12/22/13	184 KB	BayesiaLab Network File
NVES_MC2_MULTI_QUADRANT_Eco-Car	12/22/13	188 KB	BayesiaLab Network File

Although the structure of this segment-specific network is identical to that of the original network, all the relationship between nodes, factors and the target were re-estimated based on the subset of data corresponding to the *Full-Size Pickup* segment.



Relearning the Structure at the Market Level

As we move from the overall market into specific segments, and then models, we need to ask whether the structure learned at the market level will also hold true at the segment or model level.

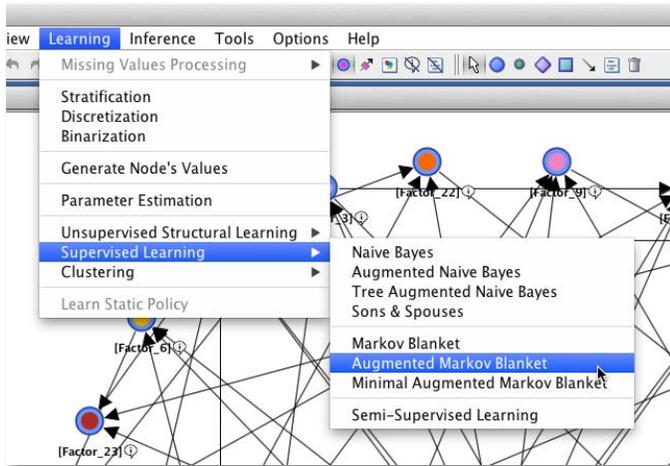
In fact, we need to make a trade-off. We can retain the richer, more complex structure learned on the basis of the entire market, and simply reestimate the parameters. Alternatively, we can relearn the network structure on the much smaller dataset of the *Full-Size Pickup* segment. As opposed to the 71,200 cases for the entire market, we would then only have 2,003 observations¹⁰ available for learning.

We hypothesize that the *Full-Size Pickup* segment does have peculiarities that lead to structural differences versus the overall market. Consequently, we decide to relearn the network structure. The number of observations we have for this segment seem adequate to learn a reliable structure.

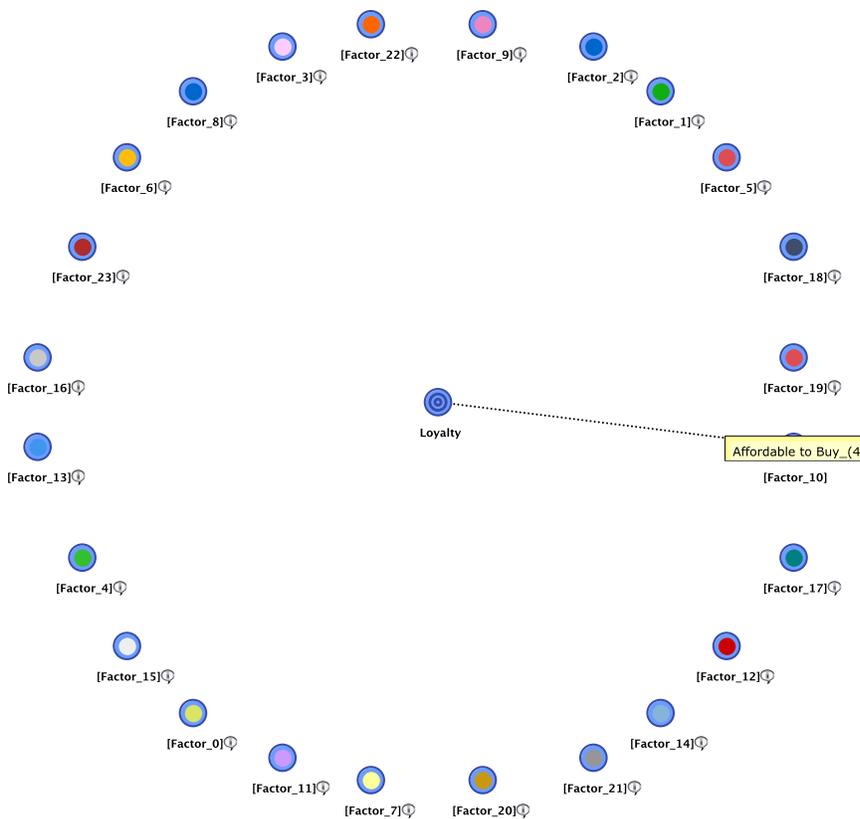
As before, we use the **Augmented Markov Blanket** algorithm: **Learning | Supervised Learning | Augmented Markov Blanket**.

¹⁰ Count of unweighted observations.

Optimizing Customer Loyalty

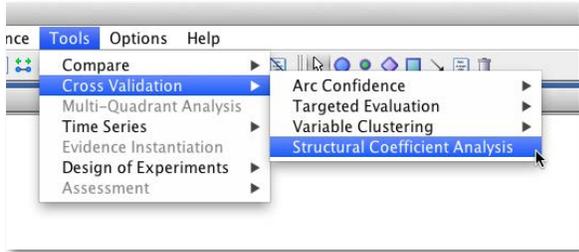


We may find the resulting network a bit surprising as only a single arc is discovered, namely a connection between *Affordable to Buy* and *Loyalty*.

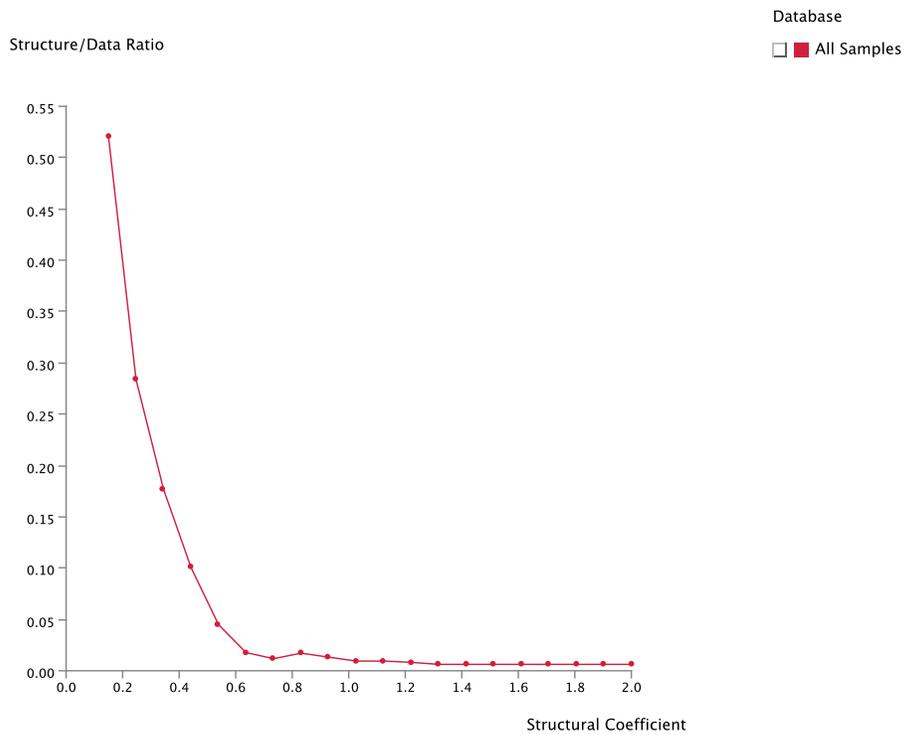


As we have not changed the default value, BayesiaLab used **SC=1** for learning. Given the smaller amount of data available for this segment, we need to examine whether this is the appropriate values here.

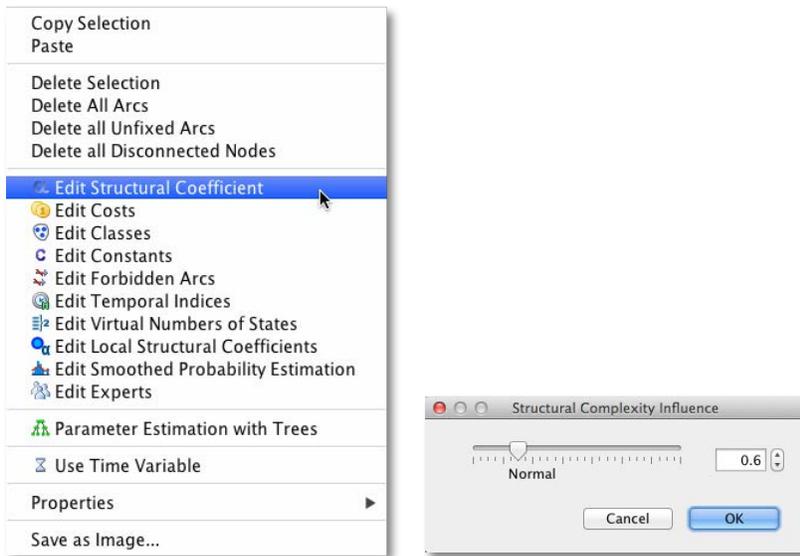
Once again, we perform a **Structural Coefficient Analysis: Tools | Cross Validation | Structural Coefficient Analysis**.



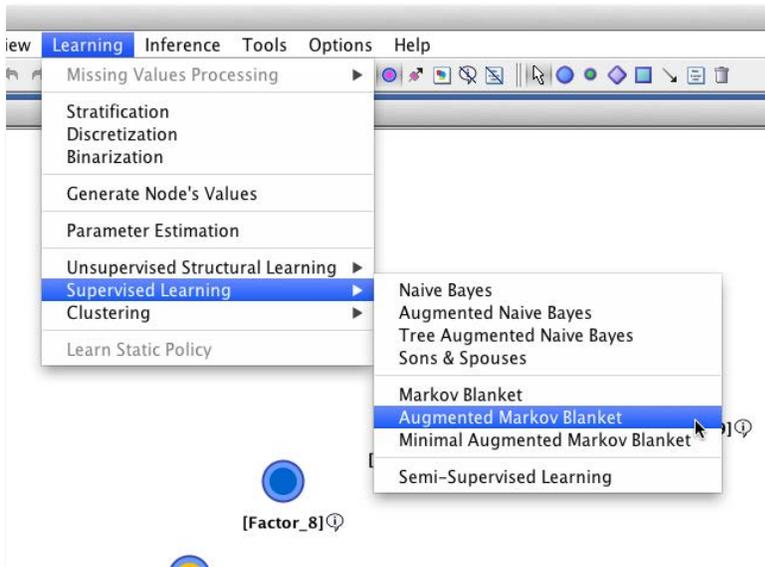
As a result, we obtain the now-familiar scree plot, which suggests that **SC=0.6** is a reasonable value



We set the **Structural Coefficient** accordingly:

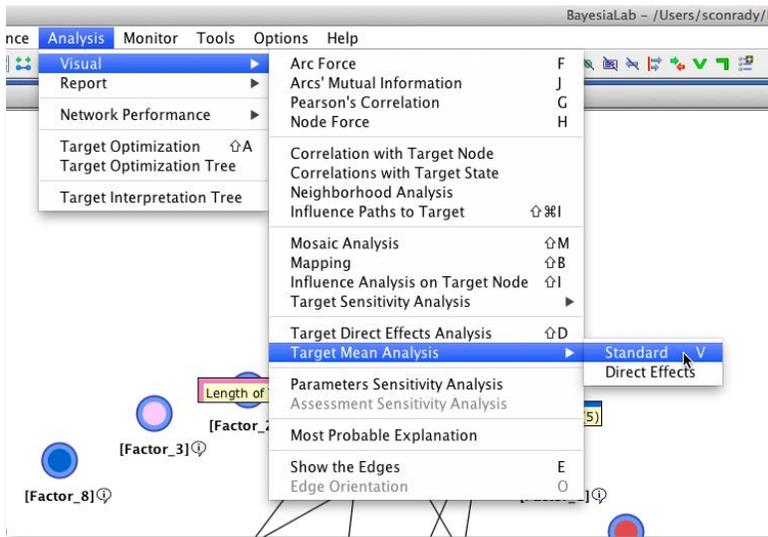


Once set, we proceed to relearning the network: **Learning | Supervised Learning | Augmented Markov Blanket.**

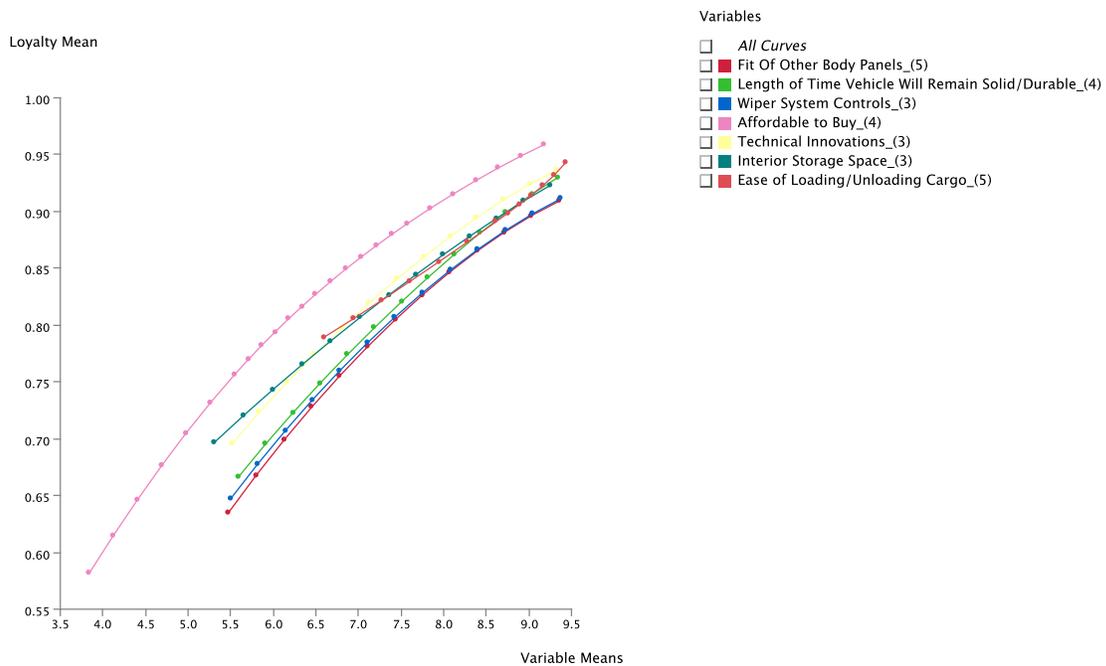


The resulting network now includes 7 factors. They appear fairly intuitive for this segment.

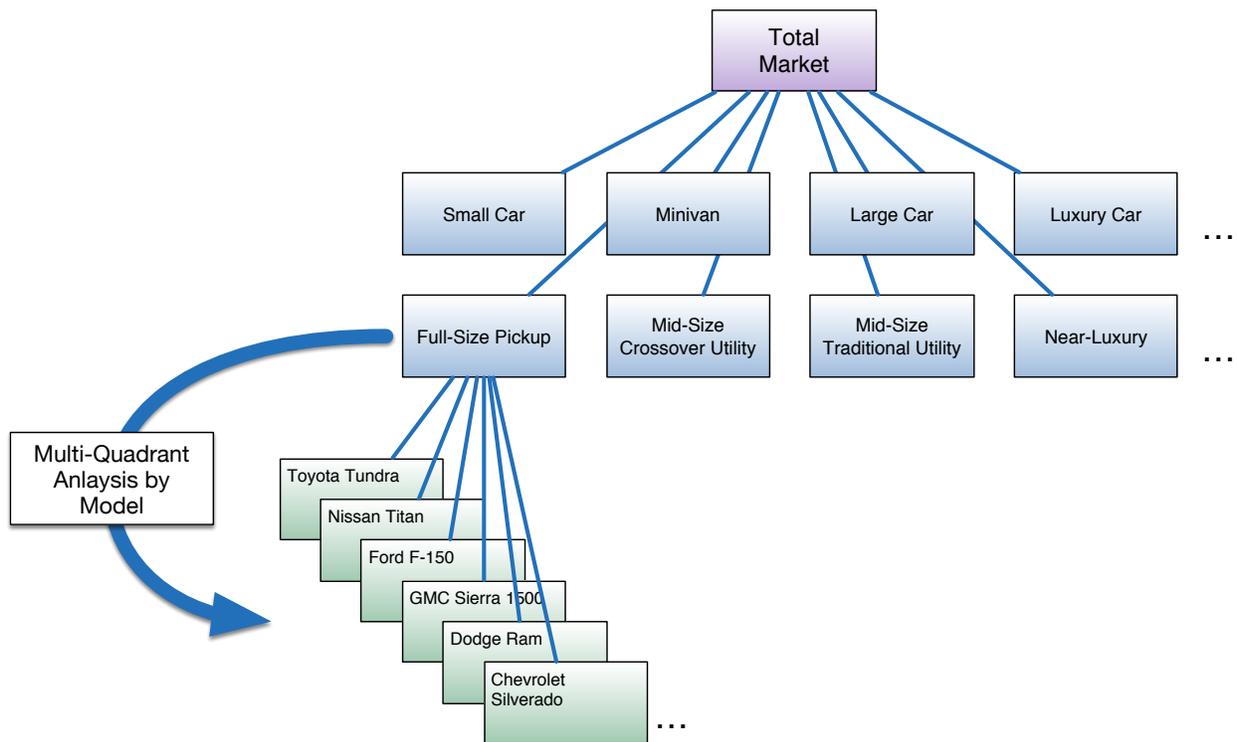
Optimizing Customer Loyalty



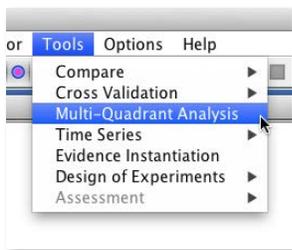
The resulting curves now show *Loyalty* as a function of the 7 factors in the network.



With the segment-specific network established, we can now proceed to the next level of detail, moving from *Segment* to *Model*.

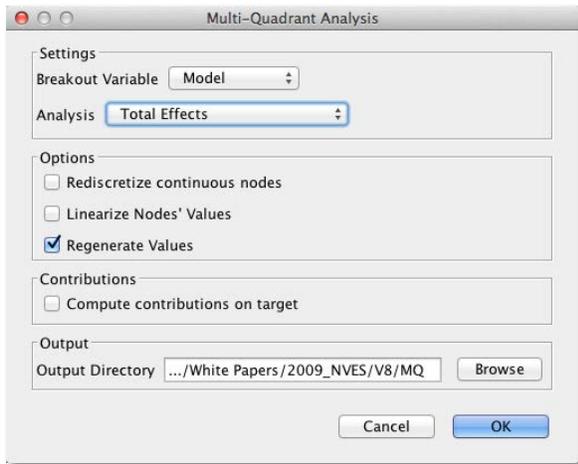


For this purpose, we rerun the **Multi-Quadrant Analysis** and select *Model* as the **Breakout Variable**.



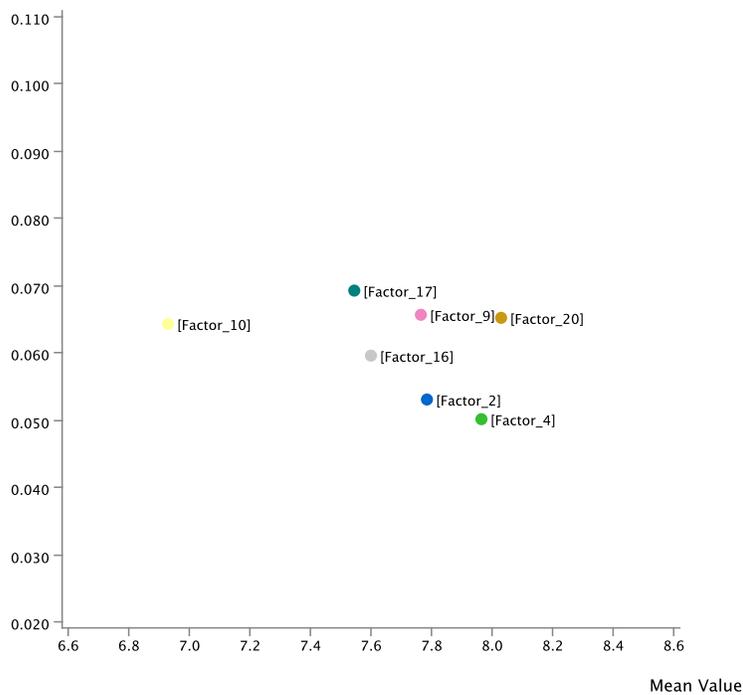
Furthermore, we must specify an output directory so we can subsequently analyze the model-specific networks.

Optimizing Customer Loyalty



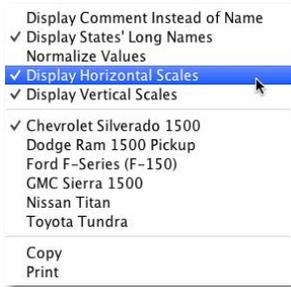
Once again, we obtain a **Quadrant Plot**, now with *Model* as the selector.

Total Effects on Loyalty (Model = Chevrolet Silverado 1500)

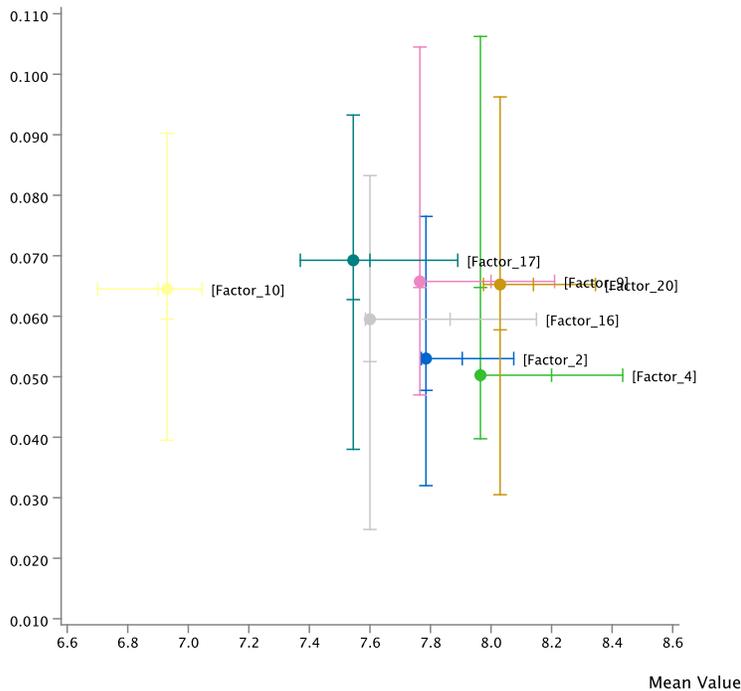


As before, we can scroll through the individual states of the selector variable. By hovering over individual variables on the plot, we see the relative position of the models with respect to any factor.

Display Horizontal Scales/Vertical Scales, which is available from the contextual menu of the **Quadrant Plot**, frames up the range of values of competitors.



Total Effects on Loyalty (Model = Chevrolet Silverado 1500)

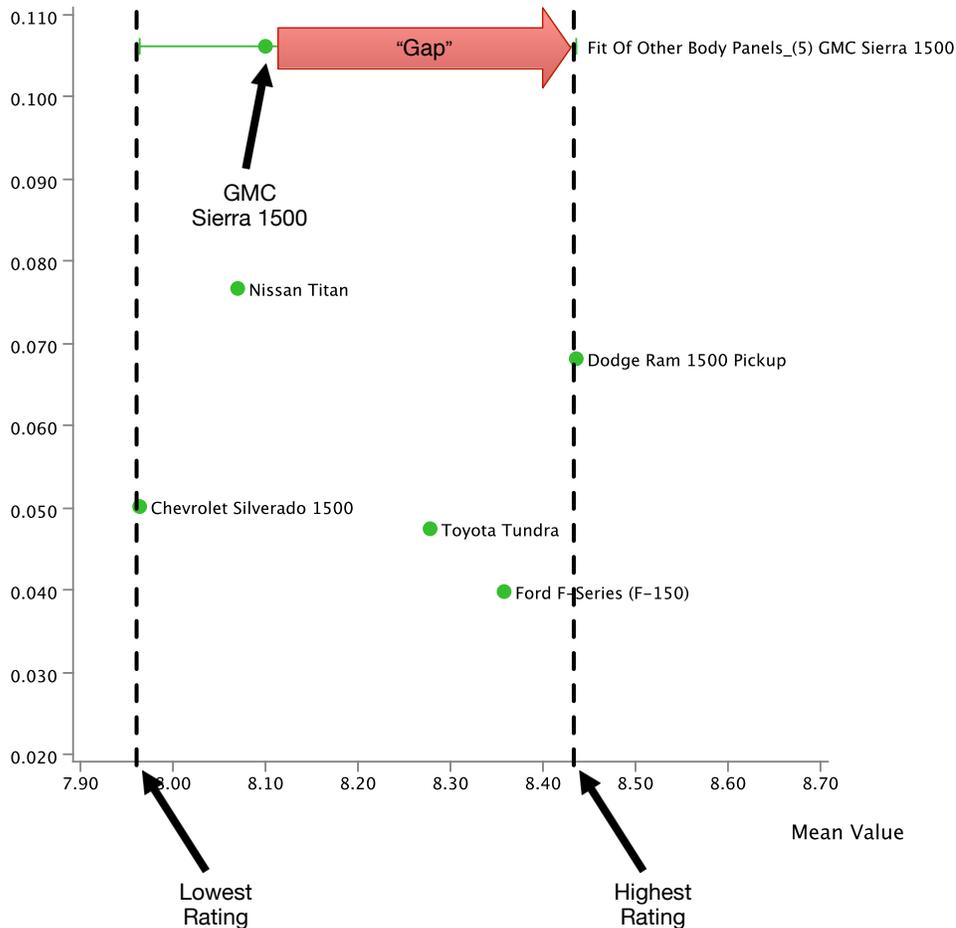


Variations within the Realm of the Possible

This view is interesting on its own; however, we can more formally utilize this information. The position of each model on the attributes' ranges allows us to compute the "gap to best/extreme level." This will subsequently become very important as we look for ways to improve brand loyalty.

For instance, with regard to *Factor 17, Fit Of Other Body Panels*, the *Dodge Ram* achieves the highest rating. For obvious reasons, this would be an appropriate target for competitors. If we were attempting to improve the rating of the *GMC Sierra*, the level of the *Dodge Ram* could be used as a target level.

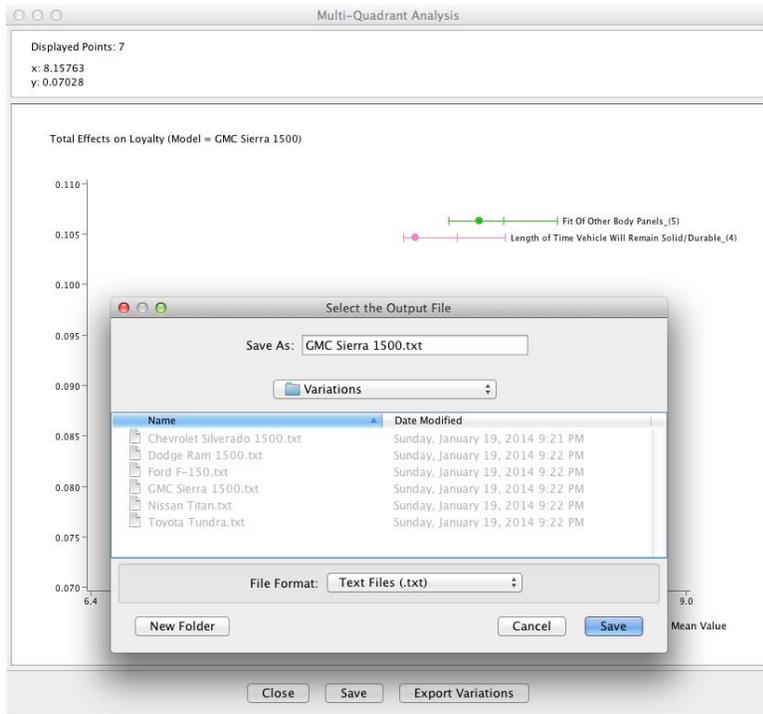
Total Effects on Loyalty (Model = GMC Sierra 1500)



As this performance level has actually been achieved, it is quite obviously feasible and thus a realistic level. This type of reference is precisely what we require for subsequent optimization. Without such a reference point, an optimization algorithm would happily recommend increasing the rating to a perfect 10, which is not at all helpful for practical purposes.

BayesiaLab can automatically extract the delta to highest and lowest levels for each factor. In this specific context, we call these deltas **Variations**. We will utilize these **Variations** as constraints for the optimization algorithm.

By clicking the **Export Variations** button, BayesiaLab saves the **Variations** for the currently selected model. For each model that we wish to optimize, we simply save this data as a text file.



Optimization

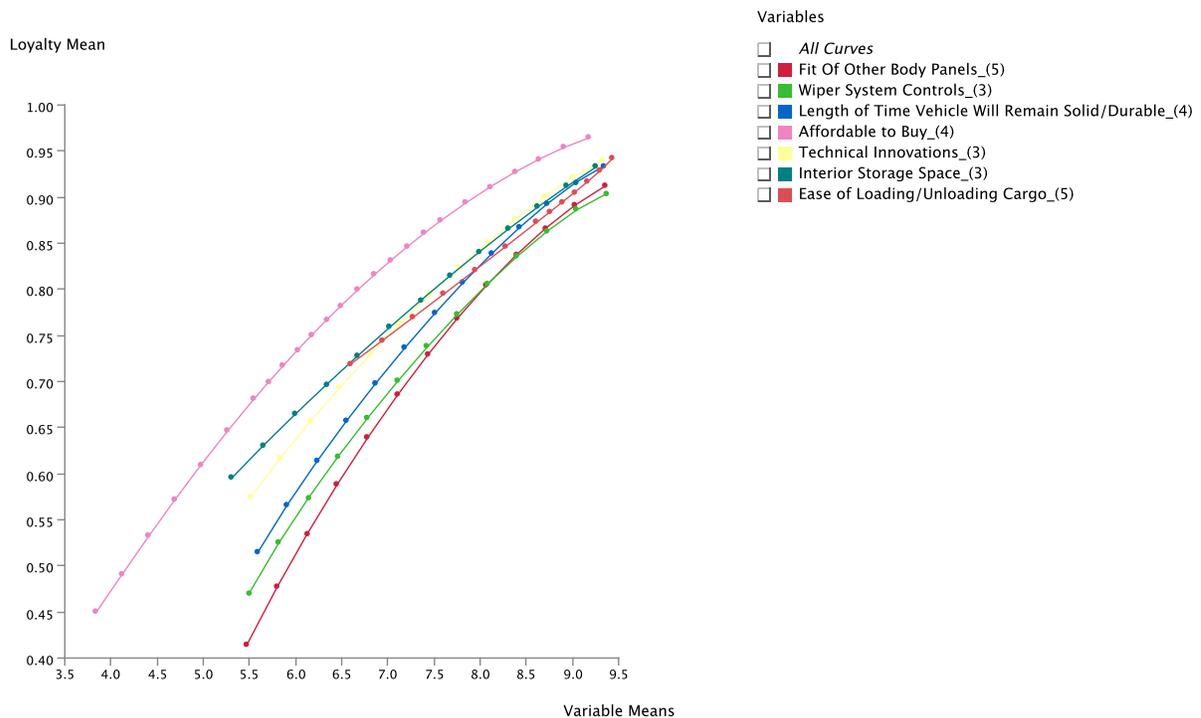
The **Multi-Quadrant Analysis** has generated new networks for each model, plus we have saved the associated **Variations**. Thus, we have all the components necessary for optimization. For the purpose of this tutorial, we will attempt to optimize the loyalty for the *GMC Sierra*.

To do so, we open the *GMC Sierra*-specific file generated with the most recent **Multi-Quadrant Analysis**.

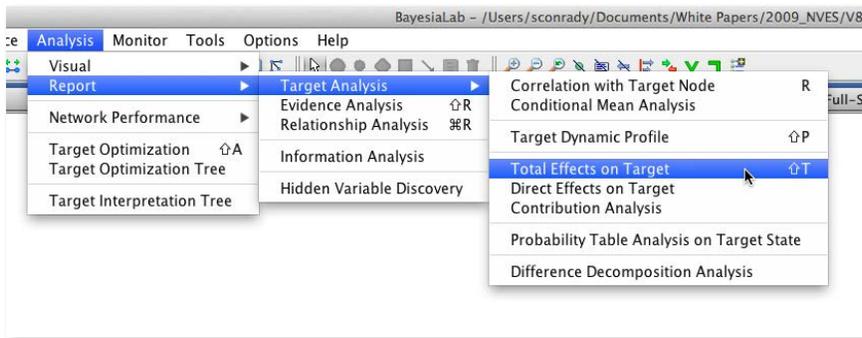
Optimizing Customer Loyalty

Name	Date Modified	Size	Kind
.DS_Store	Yesterday, 8:51 PM	6 KB	Document
NVES_MULTI_QUADRANT_Full-Size Pickup 2_MULTI_QUADRANT_Chevrolet Silverado 1500.xbl	Yesterday, 8:52 PM	17 KB	BayesiaLab Network File
NVES_MULTI_QUADRANT_Full-Size Pickup 2_MULTI_QUADRANT_Dodge Ram 1500 Pickup.xbl	Yesterday, 8:52 PM	21 KB	BayesiaLab Network File
NVES_MULTI_QUADRANT_Full-Size Pickup 2_MULTI_QUADRANT_Ford F-Series (F-150).xbl	Yesterday, 8:52 PM	20 KB	BayesiaLab Network File
NVES_MULTI_QUADRANT_Full-Size Pickup 2_MULTI_QUADRANT_GMC Sierra 1500.xbl	Yesterday, 8:52 PM	16 KB	BayesiaLab Network File
NVES_MULTI_QUADRANT_Full-Size Pickup 2_MULTI_QUADRANT_Nissan Titan.xbl	Yesterday, 8:52 PM	18 KB	BayesiaLab Network File
NVES_MULTI_QUADRANT_Full-Size Pickup 2_MULTI_QUADRANT_Toyota Tundra.xbl	Yesterday, 8:52 PM	21 KB	BayesiaLab Network File

Before proceeding to the optimization, we will briefly examine the **Target Response Functions** for the *GMC Sierra*, which we obtain via **Target Mean Analysis (Standard)**.



As earlier, when we did this at the segment level, we also run the **Total Effects on Target** report: **Analysis | Report | Target Analysis | Total Effects on Target**.

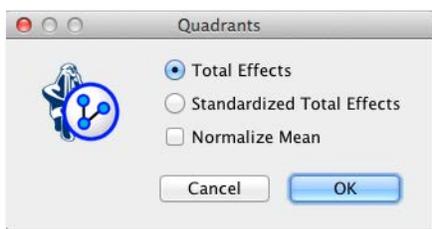


We obtain a report that shows the mean values of each factor, plus the corresponding **Total Effects**.

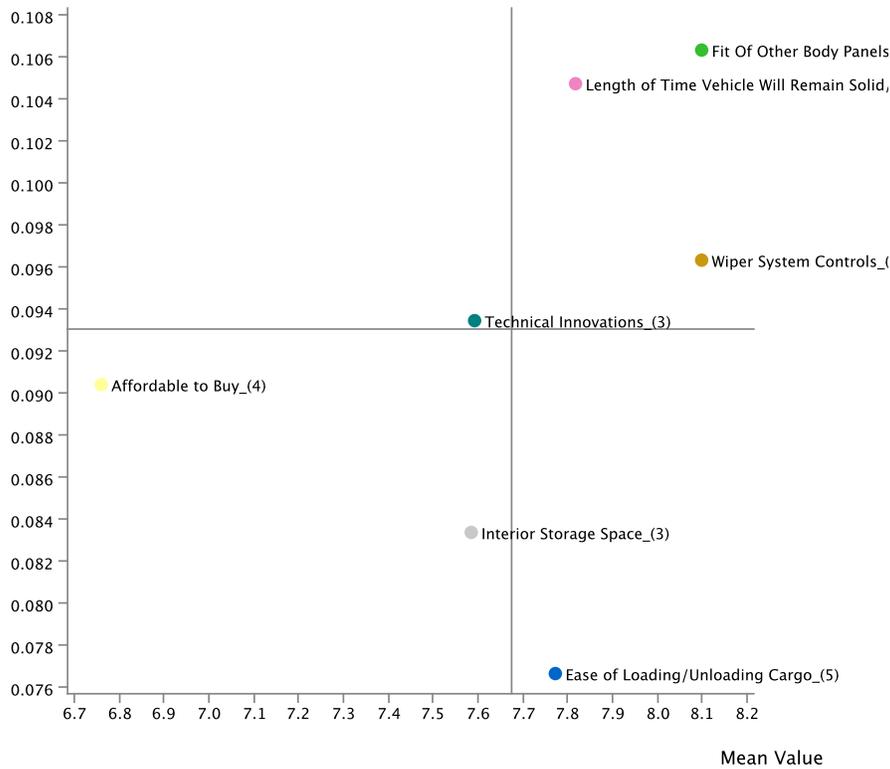
Total Effects on Target Loyalty

Node	Value/Mean	Standardized Total Effects	Total Effects	G-test	Degrees of Freedom	p-value	G-test (Data)	Degrees of Freedom (Data)	p-value (Data)
[Factor 10]	6.7630	0.6081	0.0904	101.1295	16	0.0000%	101.1295	16	0.0000%
[Factor 9]	7.8191	0.5979	0.1047	89.1921	8	0.0000%	89.1921	8	0.0000%
[Factor 4]	8.1001	0.5634	0.1064	73.6253	8	0.0000%	73.6253	8	0.0000%
[Factor 20]	8.0990	0.5508	0.0964	81.9917	8	0.0000%	77.0421	8	0.0000%
[Factor 17]	7.5954	0.5496	0.0935	71.3159	8	0.0000%	71.3159	8	0.0000%
[Factor 16]	7.5879	0.4908	0.0834	63.1153	8	0.0000%	61.6256	8	0.0000%
[Factor 2]	7.7729	0.3927	0.0767	51.6143	8	0.0000%	53.0794	8	0.0000%

Here, the **Quadrant Plot** becomes very helpful as it shows both **Value** and **Total Effects** in a single plot.



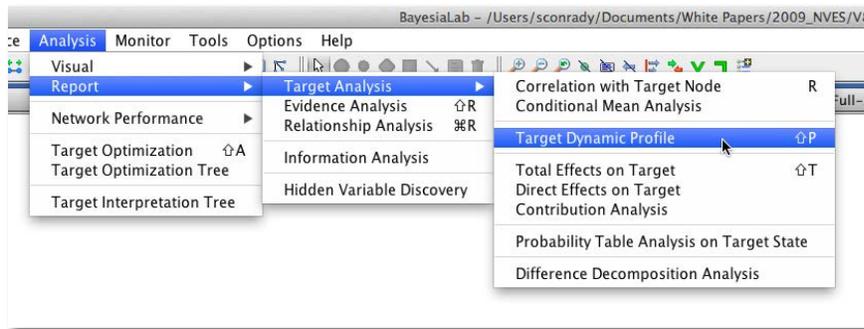
Total Effects on Loyalty



There are many ways to interpret the above plot qualitatively. For instance, we may be tempted to look at *Fit of Other Body Panels* as the top driver and suggest focusing our efforts there. Also, we might say that *Technical Innovations* is fairly important, but has room for substantial improvement.

The challenge is to determine which combination of initiatives will yield the maximum improvement for loyalty, and what the new loyalty level would be.

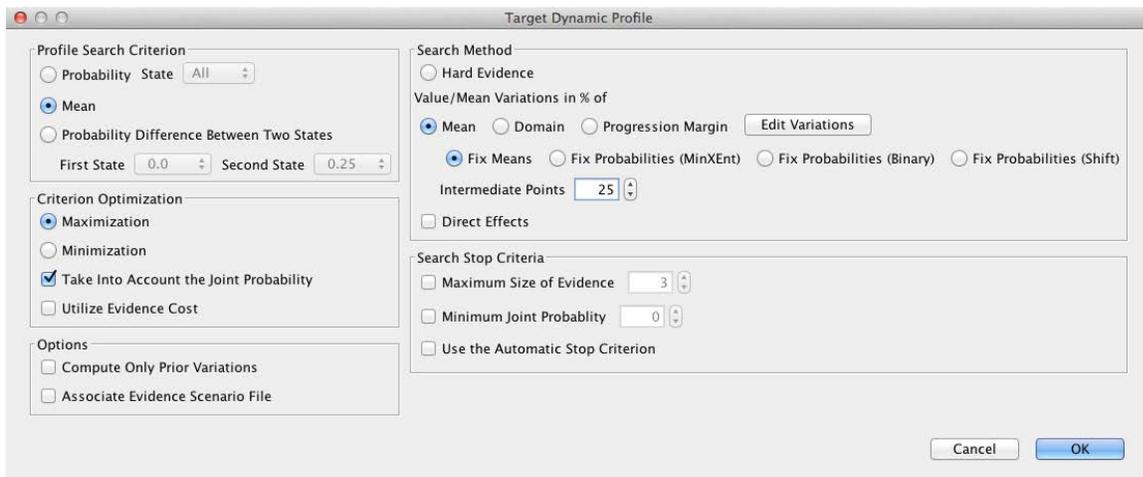
This brings us back to the very purpose of this study. We start the optimization by selecting **Analysis | Report | Target Analysis | Target Dynamic Profile:**



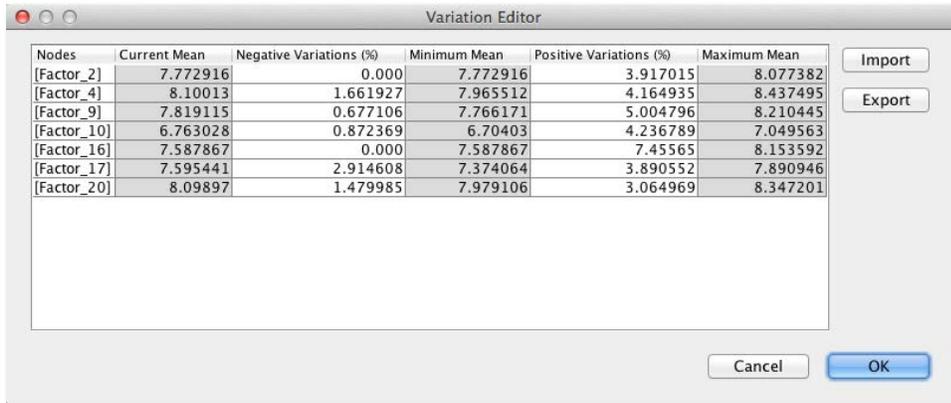
In the following window, we need to select several options that are critical in our search for optimum values. Firstly, we are looking to *maximize* the *mean* of the loyalty (rather than, for instance, maximizing the probability of certain repurchase).

The checkbox, **Take Into Account Joint Probability**, is very important in our context. As we optimize, we need to bear in mind that loyalty is expressed as a probability. We might be tempted to look for a scenario in which we obtain 100% loyalty. However, the absolute number of units to be sold as a result of loyalty is critical from a business perspective. For instance, 100% loyalty within a niche of 100 customers generates fewer sales (i.e. 100 units), than an average loyalty of 50% among a larger group of 1,000 customers (i.e. 500 units).

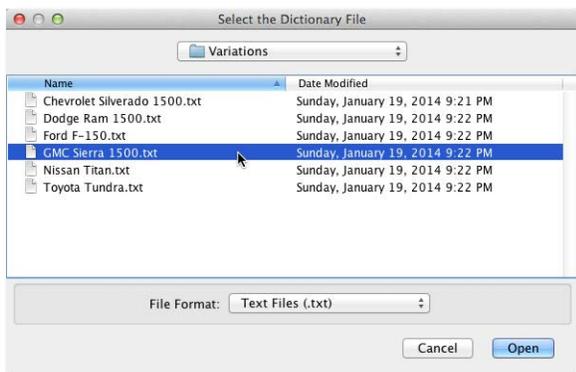
So, pursuing an idealized state of perfect loyalty may be counterproductive as it might narrow the available customer base. This is where **Joint Probability** becomes an extremely helpful concept. By virtue of having learned a Bayesian network, we automatically have the joint probability of every conceivable combination of values of all nodes. This provides us with the ability to assess how far our optimized scenarios depart from the current reality. Considering this “stretch” beyond the status quo is central to our optimization approach.



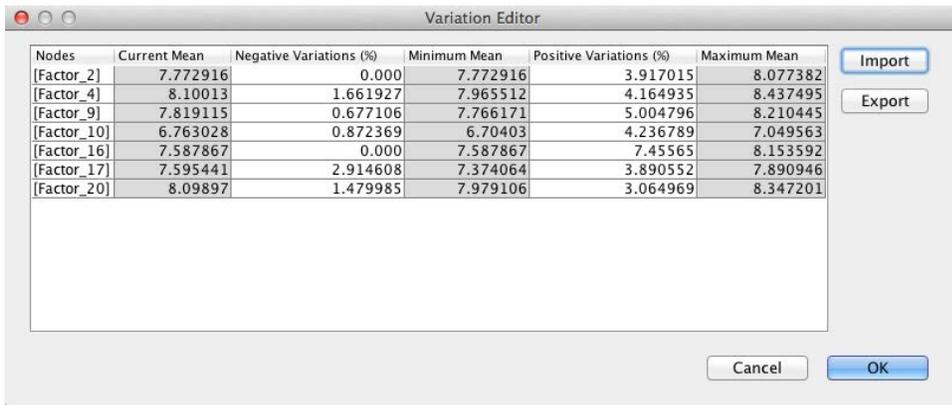
The second “reality check” relates to the variations, which we discussed earlier. By default, the **Variation Editor** is set to $\pm 100\%$. This is what we see when we first open it.



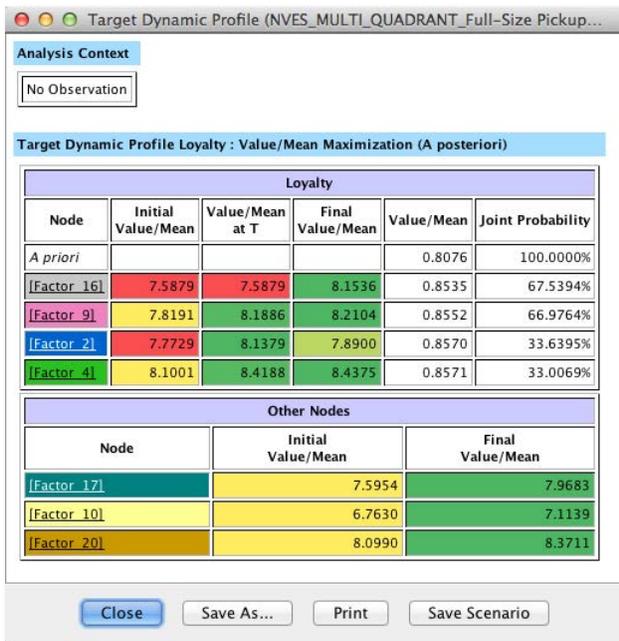
Now we re-introduce the variations we obtained earlier. By clicking **Import**, we can select the previously-saved file with the **Variations** for the *GMC Sierra*.



With the **Variations** loaded, we see the ranges within which the optimization value can search for the optimal combination of values.



Clicking **OK** immediately starts the optimization routine. Given the small size of the network, the optimization report pops up within seconds.



For a more detailed explanation, we save this report as an HTML file, which we can then open in Excel for further annotation. This file keeps all the formatting, including color-coding, of the on-screen report.

Target Dynamic Profile Loyalty : Value/Mean Maximization (A posteriori)					
Loyalty					
Node	Initial Value/Mean	Value/Mean at T	Final Value/Mean	Value/Mean	Joint Probability
A priori				0.8076	100.00%
[Factor_16]	7.5879	7.5879	8.1536	0.8535	67.54%
[Factor_9]	7.8191	8.1886	8.2104	0.8552	66.98%
[Factor_2]	7.7729	8.1379	7.89	0.857	33.64%
[Factor_4]	8.1001	8.4188	8.4375	0.8571	33.01%
Other Nodes					
Node	Initial Value/Mean	Final Value/Mean			
[Factor_17]	7.5954	7.9683			
[Factor_10]	6.763	7.1139			
[Factor_20]	8.099	8.3711			

Recommendation for GMC Sierra 1500

The above report presents the results in a highly-condensed format. It will be helpful to dissect this table cell by cell. To properly interpret this table, it should be read line-by-line, top-down.

Analysis Context	
No Observation	← No other evidence is set

Target Dynamic Profile Loyalty : Value/Mean Maximization (A posteriori)					
Loyalty					
Node	Initial Value/Mean	Value/Mean at T	Final Value/Mean	Value/Mean	Joint Probability
				Initial value of Loyalty, prior to optimization ↓	Initial joint probability is 100% ↓
<i>A priori</i>				0.8076	100.00%
[Factor 16]	7.5879	7.5879	8.1536	0.8535	67.54%
↑ Most important factor	↑ Initial value of Factor 16, prior to optimization		↑ Optimal value of Factor 16, within given constraints.	↑ Value of Loyalty, after Factor 16 is set to optimal value	↑ New joint probability after setting Factor 16. This means that 67.54% of the observations already meet this condition.
[Factor 9]	7.8191	8.1886	8.2104	0.8552	66.98%
↑ 2nd most important factor	↑ Initial value of Factor 9, prior to optimization	↑ Value of Factor 9, after Factor 16 is set to optimal value	↑ Optimal value of Factor 9, within given constraints.	↑ Value of Loyalty, after Factor 9 is set to optimal value	↑ New joint probability after setting Factor 16
[Factor 2]	7.7729	8.1379	7.89	0.857	33.64%
↑ 3rd most important factor	↑ Initial value of Factor 2, prior to optimization	↑ Value of Factor 2, after Factor 9 is set to optimal value	↑ Optimal value of Factor 2, within given constraints.	↑ Value of Loyalty, after Factor 2 is set to optimal value	↑ New joint probability after setting Factor 9
[Factor 4]	8.1001	8.4188	8.4375	0.8571	33.01%
↑ 4th most important factor	↑ Initial value of Factor 4, prior to optimization	↑ Value of Factor 4, after Factor 9 is set to optimal value	↑ Optimal value of Factor 4, within given constraints.	↑ Value of Loyalty, after Factor 4 is set to optimal value	↑ New joint probability after setting Factor 2

Factors listed in order of priority

Decreasing joint probability indicates departure from status quo, i.e. departure from current reality

↑ This means that after applying all four listed measures, an increase of 0.05, i.e. 5 percentage points, would be observed.

Other Nodes		
Node	Initial Value/Mean	Final Value/Mean
[Factor 17]	7.5954	7.9683
	↑ Initial value of Factor 17, prior to optimization	↑ Final value of Factor 17, after setting the top-4 factors to the optimal levels.
[Factor 10]	6.763	7.1139
	↑ Initial value of Factor 10, prior to optimization	↑ Final value of Factor 10, after setting the top-4 factors to the optimal levels.
[Factor 20]	8.099	8.3711
	↑ Initial value of Factor 20, prior to optimization	↑ Final value of Factor 20, after setting the top-4 factors to the optimal levels.

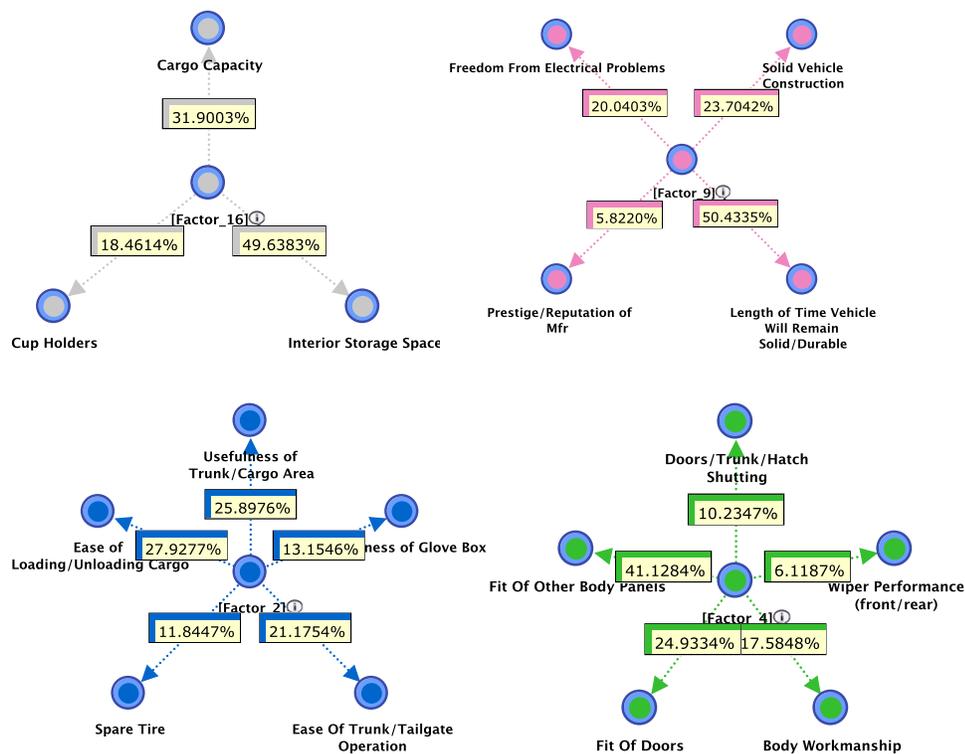
Collateral Effects

Even though changing the remaining factors can no further improve Loyalty, these nodes are affected by the changes of the other factors. The changes can be interpreted as collateral effects.

This table clearly spells out the top priorities for the *GMC Sierra 1500*. According to this simulation, achieving the new levels of the factors would lift loyalty from 0.80 to 0.85. For the GMC, this would translate into several thousand more customers returning to the brand.

For reference, we highlight the manifest nodes that make up the factors:

Optimizing Customer Loyalty



Given that the earlier **Multi-Quadrant Analysis** generated networks for all models in this segment, we could now repeat the optimization for any of the other models within minutes.

Summary

Bayesian networks and BayesiaLab make it possible to identify relevant drivers from previously-indistinguishable product ratings in survey data. BayesiaLab can perform optimization on that basis and immediately establish priorities. With this approach, market researchers can—quickly and transparently—generate clear recommendations for decision makers.

Appendix

Variables

Select Variables from the 2009 Strategic Vision New Vehicle Experience Survey (NVES)

Combined Base Weight	Door Handles – Exterior	Fuel Efficiency
Segment	Badging (Exterior logos/identifiers)	Emissions Control
Make	Exterior Color	Front Seat Roominess
Model	Headlights Design	2nd row Seat Roominess
Loyalty	Taillights Design	Ease of Front Seat Entry
Safety Features	Sunroof	Ease of 2nd Row Seat Entry
Front Visibility – Driver	Interior Colors	Comfort of Seatbelts
Rear Visibility – Driver	Interior Trim & Finish	Support of Seats
Braking	Body Workmanship	Passenger Seating Capacity
Headlights Function	Fit Of Doors	Interior Storage Space
Taillights Function	Fit Of Other Body Panels	Cargo Capacity
Turn Signal Function	Doors/Trunk/Hatch Shutting	Cup Holders
Airbags	Wiper Performance (front/rear)	Ease Of Trunk/Tailgate Operation
Bumpers	Quality Of Interior Materials	Ease of Loading/Unloading Cargo
Solid Vehicle Construction	Instrument Cluster Gauges	Front Seat Comfort
Length of Time Vehicle Will Remain Solid/ Durable	Door Handles – Interior	2nd row Seat Comfort
Freedom From Electrical Problems	Interior Lighting	Driver Seat Adjustability
Ground Clearance	Quality of Seat Materials	Passenger Seat Adjustability
Riding Comfort	Freedom From Squeaks And Rattles	Driver Armrests
Maneuverability	Freedom From Engine Noise	Seating Versatility
Turning Radius	Freedom From Road Noise	Seating Stowaway/Conversion
Road Holding Ability	Freedom From Wind Noise	Placement Of Controls/Instruments
Handling	Smoothness At Idle	Electronic Display of Information
Steering Feedback	Smoothness Of Transmission	Ease of Reading Controls/Instruments
Acceleration From Stop	Window Controls	Usefulness of Glove Box
Passing Capability	Wiper System Controls	Usefulness of Trunk/Cargo Area
Lines/Flow of the Vehicle	Speed Control System	Spare Tire
Appearance Of Paint Job	Speakers	Price Or Deal Offered
Size/Proportions	Ability to Control Sound Quality	Future Trade-In Or Resale Value
Side Mirrors	Sound System Controls	Warranty Coverage
Appearance of Wheels & Rims	CD Player	Technical Innovations
Appearance of Tires	Operation of HVAC Controls	Level of Standard Equipment
Exterior Trim/Molding	HVAC Vents	Fuel Economy/Mileage
	Defrost/Defog	Economical to Own

List of Factors

Classes	Nodes
[Factor_0]	Door Handles - Interior
	Ease of Reading Controls/Instruments
	Electronic Display of Information
	Instrument Cluster Gauges
	Interior Lighting
	Placement Of Controls/Instruments
	Sunroof
[Factor_1]	Freedom From Engine Noise
	Freedom From Road Noise
	Freedom From Squeaks And Rattles
	Freedom From Wind Noise
	Smoothness At Idle
[Factor_2]	Smoothness Of Transmission
	Ease Of Trunk/Tailgate Operation
	Ease of Loading/Unloading Cargo
	Spare Tire
[Factor_3]	Usefulness of Glove Box
	Usefulness of Trunk/Cargo Area
	Driver Armrests
	Driver Seat Adjustability
[Factor_4]	Passenger Seat Adjustability
	Seating Stowaway/Conversion
	Seating Versatility
	Body Workmanship
	Doors/Trunk/Hatch Shutting
[Factor_5]	Fit Of Doors
	Fit Of Other Body Panels
	Wiper Performance (front/rear)
	Airbags
	Bumpers
[Factor_6]	Front Visibility - Driver
	Rear Visibility - Driver
	Safety Features
	2nd row Seat Roominess
	Ease of 2nd Row Seat Entry
[Factor_7]	Ease of Front Seat Entry
	Front Seat Roominess
	Passenger Seating Capacity
	Appearance of Tires
	Appearance of Wheels & Rims
[Factor_8]	Badging (Exterior logos/identifiers)
	Door Handles - Exterior
	Exterior Trim/Molding
	Ground Clearance
	Handling
[Factor_9]	Riding Comfort
	Road Holding Ability
	Steering Feedback
	Freedom From Electrical Problems
[Factor_10]	Length of Time Vehicle Will Remain Solid/Durable
	Prestige/Reputation of Mfr
	Solid Vehicle Construction
	Affordable to Buy
[Factor_11]	Economical to Own
	Future Trade-In Or Resale Value
	Price Or Deal Offered
	Appearance Of Paint Job
[Factor_12]	Lines/Flow of the Vehicle
	Side Mirrors
	Size/Proportions
	Interior Colors
[Factor_13]	Interior Trim & Finish
	Quality Of Interior Materials
	Quality of Seat Materials
	Braking
[Factor_14]	Headlights Function
	Taillights Function
	Turn Signal Function
	2nd row Seat Comfort
[Factor_15]	Comfort of Seatbelts
	Front Seat Comfort
	Support of Seats
	Ability to Control Sound Quality
[Factor_16]	CD Player
	Sound System Controls
	Speakers
	Cargo Capacity
[Factor_17]	Cup Holders
	Interior Storage Space
	Level of Standard Equipment
	Technical Innovations
[Factor_18]	Warranty Coverage
	Exterior Color
	Headlights Design
	Taillights Design
[Factor_19]	Emissions Control
	Fuel Economy/Mileage
	Fuel Efficiency
	Speed Control System
[Factor_20]	Window Controls
	Wiper System Controls
	Defrost/Defog
	HVAC Vents
[Factor_21]	Operation of HVAC Controls
	Maneuverability
	Turning Radius
	Acceleration From Stop
[Factor_22]	Passing Capability

Contact Information

Bayesia USA

312 Hamlet's End Way

Franklin, TN 37067

USA

Phone: +1 888-386-8383

info@bayesia.us

www.bayesia.us

Bayesia Singapore Pte. Ltd.

20 Cecil Street

#14-01, Equity Plaza

Singapore 049705

Phone: +65 3158 2690

info@bayesia.sg

www.bayesia.sg

Bayesia S.A.S.

6, rue Léonard de Vinci

BP 119

53001 Laval Cedex

France

Phone: +33(0)2 43 49 75 69

info@bayesia.com

www.bayesia.com

Copyright

© 2014 Bayesia USA, Bayesia S.A.S. and Bayesia Singapore Pte. Ltd. All rights reserved.