

---

# **Data Mining**

## **SPSS Clementine 12.0**

### **4. Handling Missing and Outliers Values**

**Spring 2010**  
Instructor: Dr. Masoud Yaghini

# Outline

---

- **Overview**
- **Add A Source Node**
- **Add A Type Node**
- **Add A Data Audit Node**
- **Browsing Statistics and Charts**
- **Handling Missing Values**
- **Handling Outliers Values**
- **References**

---

# Overview

# Overview

---

- The **Data Audit** node provides a comprehensive first look at the data you bring into Clementine.
- **Data Audit** node often used during the initial data exploration
- The data audit report shows
  - summary statistics
  - histograms
  - distribution graphs for each data field
- It allows you to specify treatments for missing values, outliers, and extreme values.

# Overview

---

- This example uses:
  - The stream named *telco\_dataaudit.str*
  - The data file named *telco.sav*.
- These files are available from the Demos directory of any Clementine Client installation.
- The *telco\_dataaudit.str* file is in the *Segmentation\_Module* directory.
- The example focuses on using demographic data to predict usage patterns.

---

# **Add A Source Node**

# Building Source Node

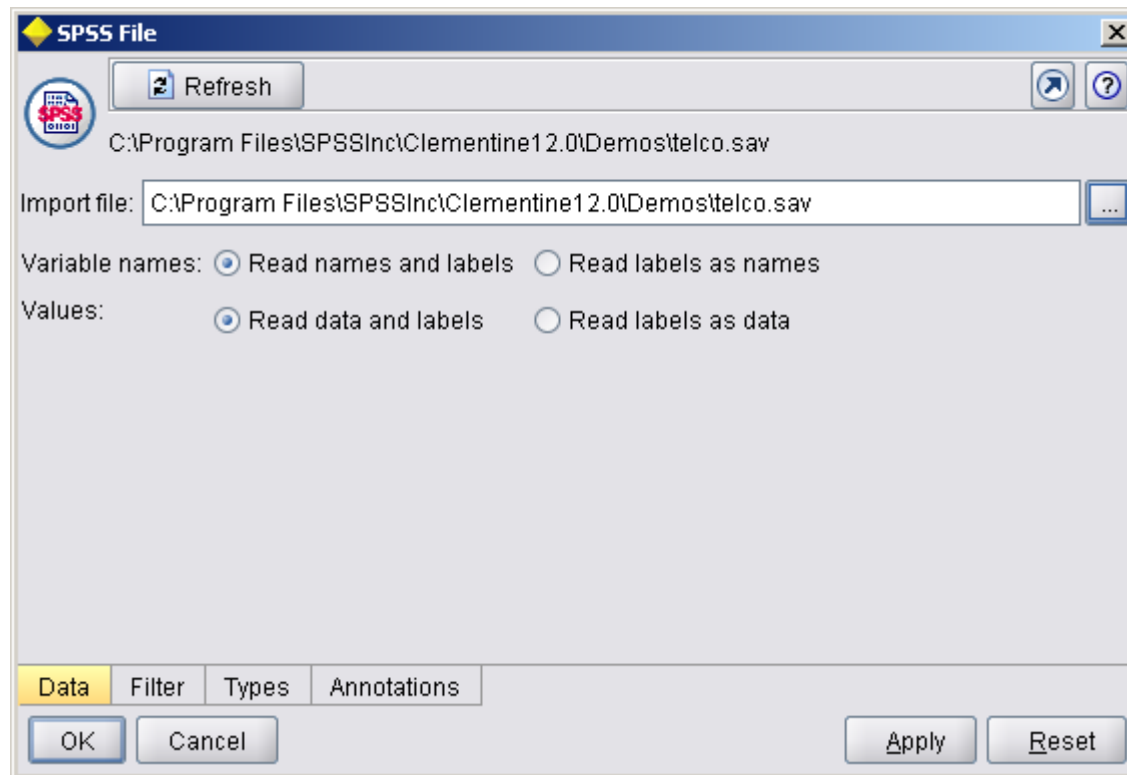
---

- Add an **SPSS** source node



# Building Source Node

- Pointing to *telco.sav*.





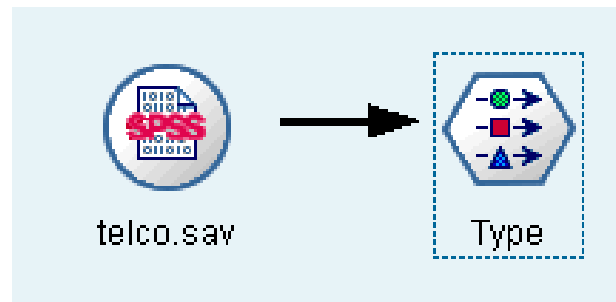
---

# **Add A Type Node**

# Building Type Node

---

- Add a **Type** node to define fields, and



# Data Type

---

- Field properties can be specified in a source node or in a separate **Type** node.
- **Data Type**
  - describes the usage of the data fields in Clementine.
  - Used to describe characteristics of the data in a given field.
  - If all of the details of a field are known, it is called **fully instantiated**.
  - The type of a field is different from the storage of a field, which indicates whether data are stored as strings, integers, real numbers, dates, times, or timestamps.
  - For example, you may want to set the type for an integer field with values of 1 and 0 to flag. This usually indicates that 1 = *True* and 0 = *False*.

# Data Type

---

- The following data types are available:
  - **Range**
    - ◆ Used to describe numeric values, such as a range of 0–100 or 0.75–1.25.
    - ◆ A range value can be an **integer**, **real number**, or **date/time**.
  - **Discrete**
    - ◆ Used for **string values** when an exact number of distinct values is unknown.
    - ◆ This is an uninstantiated data type, meaning that all possible information about the storage and usage of the data is not yet known.
    - ◆ Once data have been read, the type will be flag, set, or typeless, depending on the **maximum set size** specified in the stream properties dialog box.

# Data Type

---

## — Flag

- ◆ Used for data with two distinct values, such as Yes and No or 1 and 2.
- ◆ Data may be represented as text, integer, real number, or date/time.
- ◆ Note: Date/time refers to three types of storage: time, date, or timestamp.

## — Set

- ◆ Used to describe data with multiple distinct values, each treated as a member of a set, such as small/medium/large.
- ◆ A set can have any storage—numeric, string, or date/time.
- ◆ Note that setting a type to **Set** does not automatically change the values to string.

# Data Type

---

## — Ordered Set

- ◆ Used to describe data with multiple distinct values that have an order.
- ◆ For example, salary categories or satisfaction rankings can be typed as an ordered set.
- ◆ The order of an ordered set is defined by the natural sort order of its elements.
- ◆ For example, 1, 3, 5 is the default sort order for a set of integers, while HIGH, LOW, NORMAL (ascending alphabetically) is the order for a set of strings.
- ◆ The ordered set type enables you to define a set of categorical data as ordinal data for the purposes of visualization, model building (C5.0, C&R Tree, TwoStep), and export to other applications, such as SPSS, that recognize ordinal data as a distinct type.
- ◆ You can use an ordered set field anywhere that a set field can be used.
- ◆ The fields of any storage type (real, integer, string, date, time, and so on) can be defined as an ordered set.

# Data Type

---

## — Typeless

- ◆ Used for data that does not conform to any of the above types or for set types with too many members.
- ◆ It is useful for cases in which the type would otherwise be a set with many members (such as an account number).
- ◆ When you select **Typeless** for a field, the role is automatically set to None.
- ◆ The default maximum size for sets is 250 unique values.
- ◆ This number can be adjusted or disabled in the stream properties dialog box.

# Data Type

---

- You can manually specify data types, or you can allow the software to read the data and determine the type based on the values that it reads.
- **To Use Auto-Typing**
  - In a **Type** node or the **Types** tab of a source node, set the Values column to <Read> for the desired fields.
    - ◆ This will make metadata available to all nodes downstream.
    - ◆ You can quickly set all fields to <Read> or <Pass> using the sunglasses buttons on the dialog box.
  - Click **Read Values** to read values from the data source immediately.



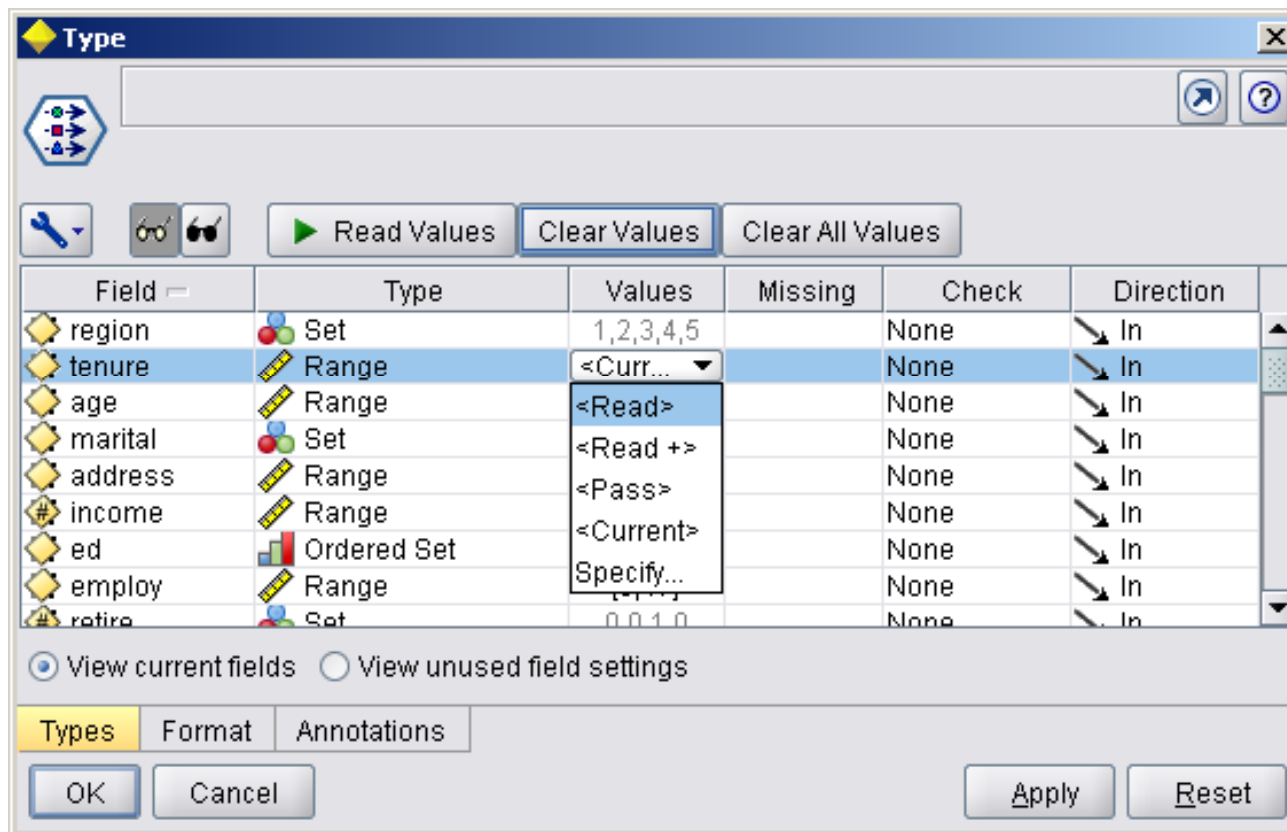
# Data Type

---

- To Manually Set the **Type** for a Field
  - Select a field in the table.
  - From the drop-down list in the Type column, select a type for the field.
  - Alternatively, you can use Ctrl-A or Ctrl-click to select multiple fields before using the drop-down list to select a type.

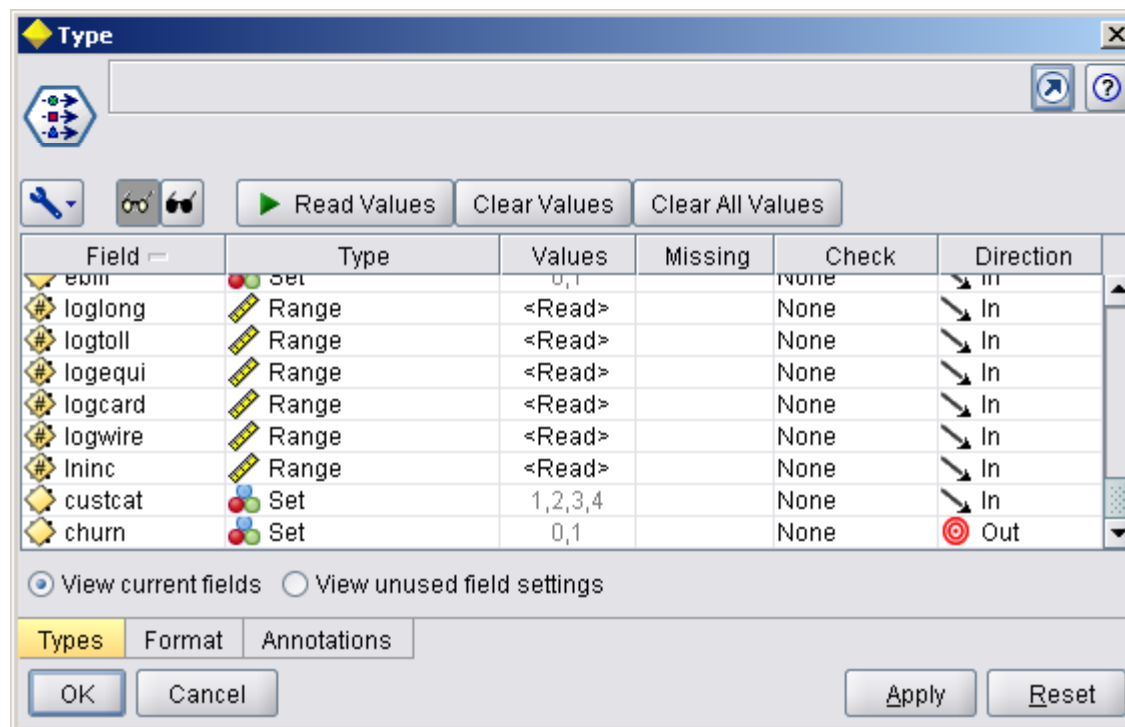
# Data Type

- Type node



# Directions

- Specify *churn* as the target field (**Direction = Out**).
- **Direction** should be set to **In** for all of the other fields so that this is the only target.

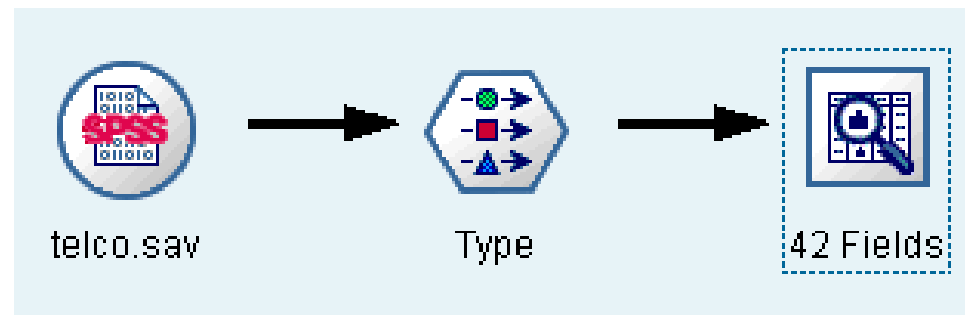


---

# **Add A Data Audit Node**

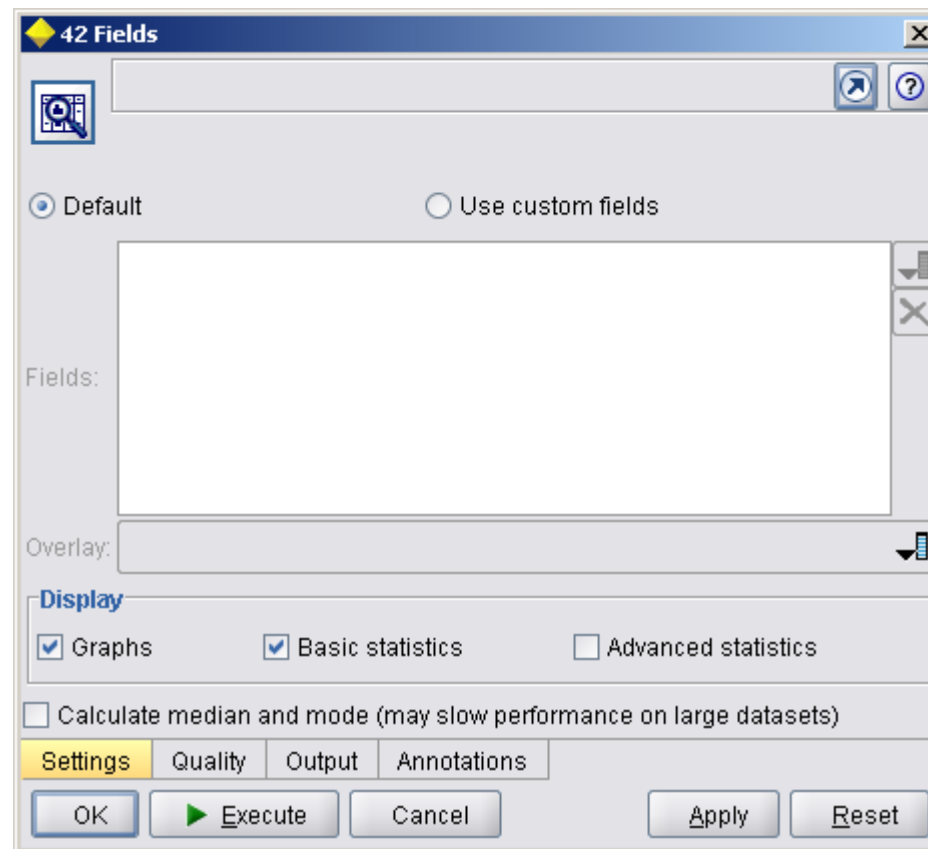
# Building the Stream

- Attach a **Data Audit** node to the stream.



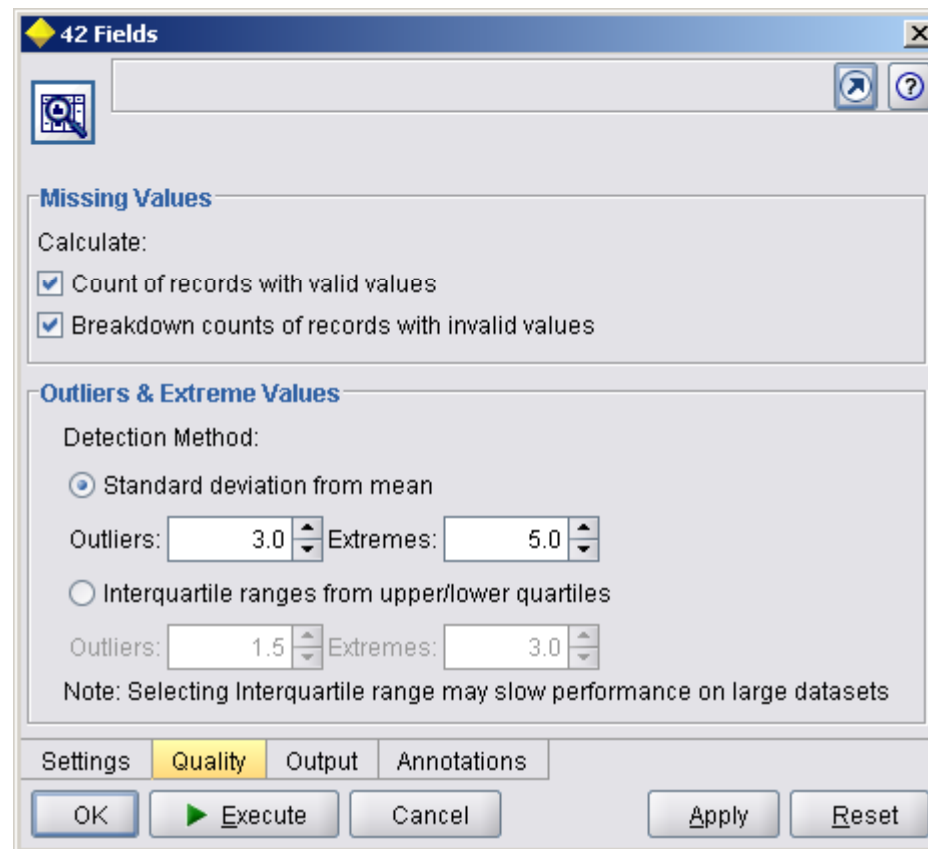
# Building the Stream

- On the **Settings** tab, leave the default settings in place to include all fields in the report.



# Building the Stream

- On the **Quality** tab, leave the default settings for detecting missing values, outliers, and extreme values in place, and click **Execute**.



# Building the Stream

---

- **Data Audit Quality Tab**

- **Missing Values**

- ◆ Count of records with valid values. Select this option to show the number of records with valid values for each evaluated field.
    - ◆ Note that null (undefined) values, blank values, white spaces and empty strings are always treated as invalid values.

- **Breakdown counts of records with invalid values**

- ◆ Select this option to show the number of records with each type of invalid value for each field.



---

## ● Data Audit Quality Tab

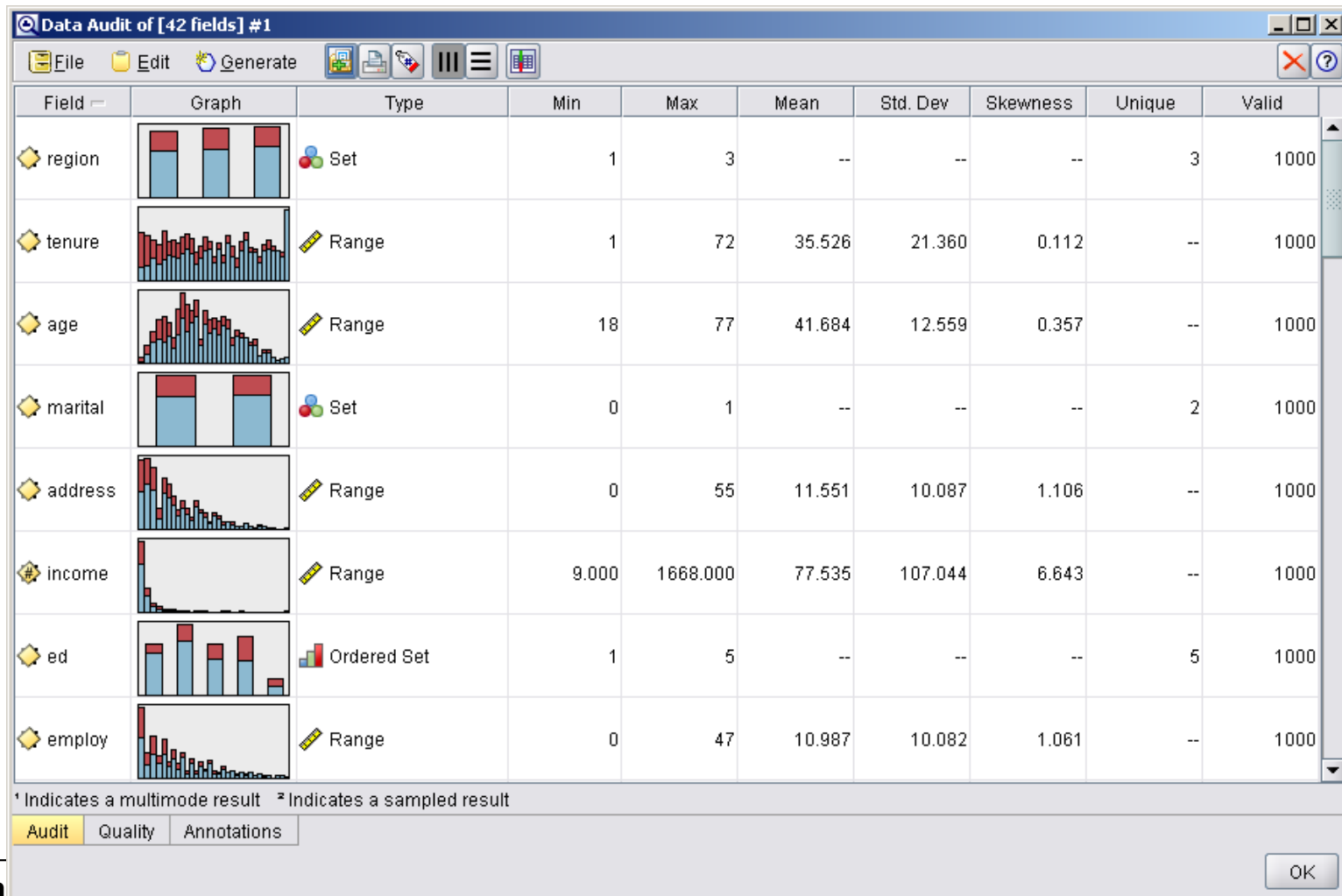
- Standard deviation from the mean.
  - ◆ Detects outliers and extremes based on the number of standard deviations from the mean.
  - ◆ For example, if you have a field with a mean of 100 and a standard deviation of 10, you could specify 3.0 to indicate that any value below 70 or above 130 should be treated as an outlier.
- Interquartile range.
  - ◆ Detects outliers and extremes based on the interquartile range, which is the range within which the two central quartiles fall (between the 25th and 75th percentiles).
  - ◆ For example, based on the default setting of 1.5, the lower threshold for outliers would be  $Q1 - 1.5 * IQR$  and the upper threshold would be  $Q3 + 1.5 * IQR$ .
  - ◆ Note that using this option may slow performance on large datasets.

---

# Browsing Statistics and Charts

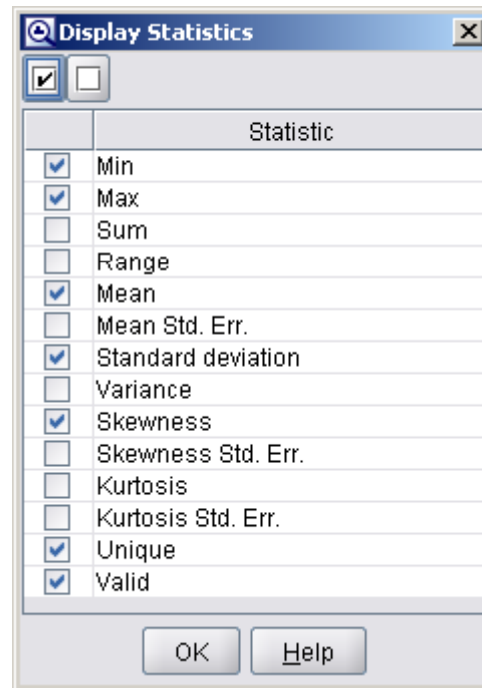
# Browsing Statistics and Charts

- The **Data Audit** browser is displayed, with thumbnail graphs and descriptive statistics for each field.



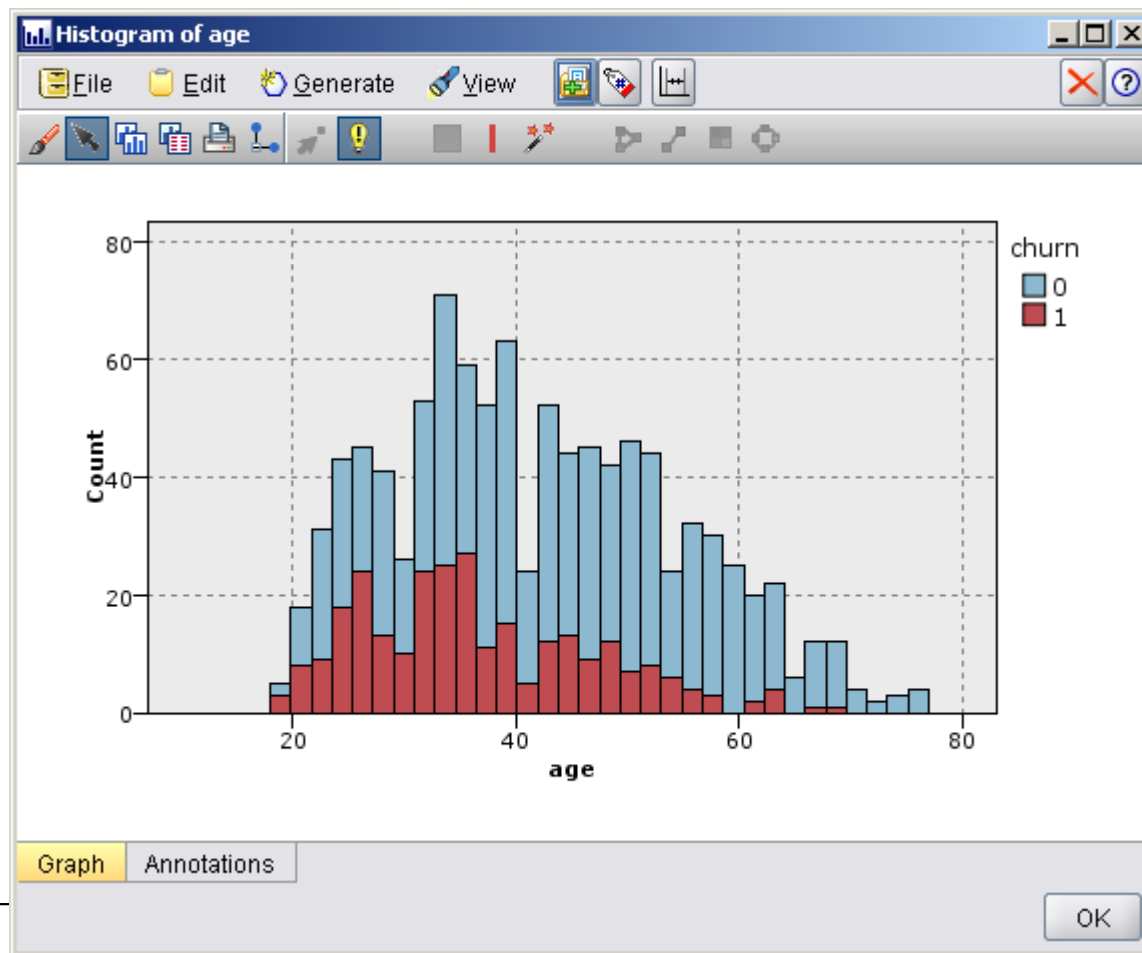
# Browsing Statistics and Charts

- You can also use the toolbar or **Edit > Display statistics** menu to choose the statistics to display.



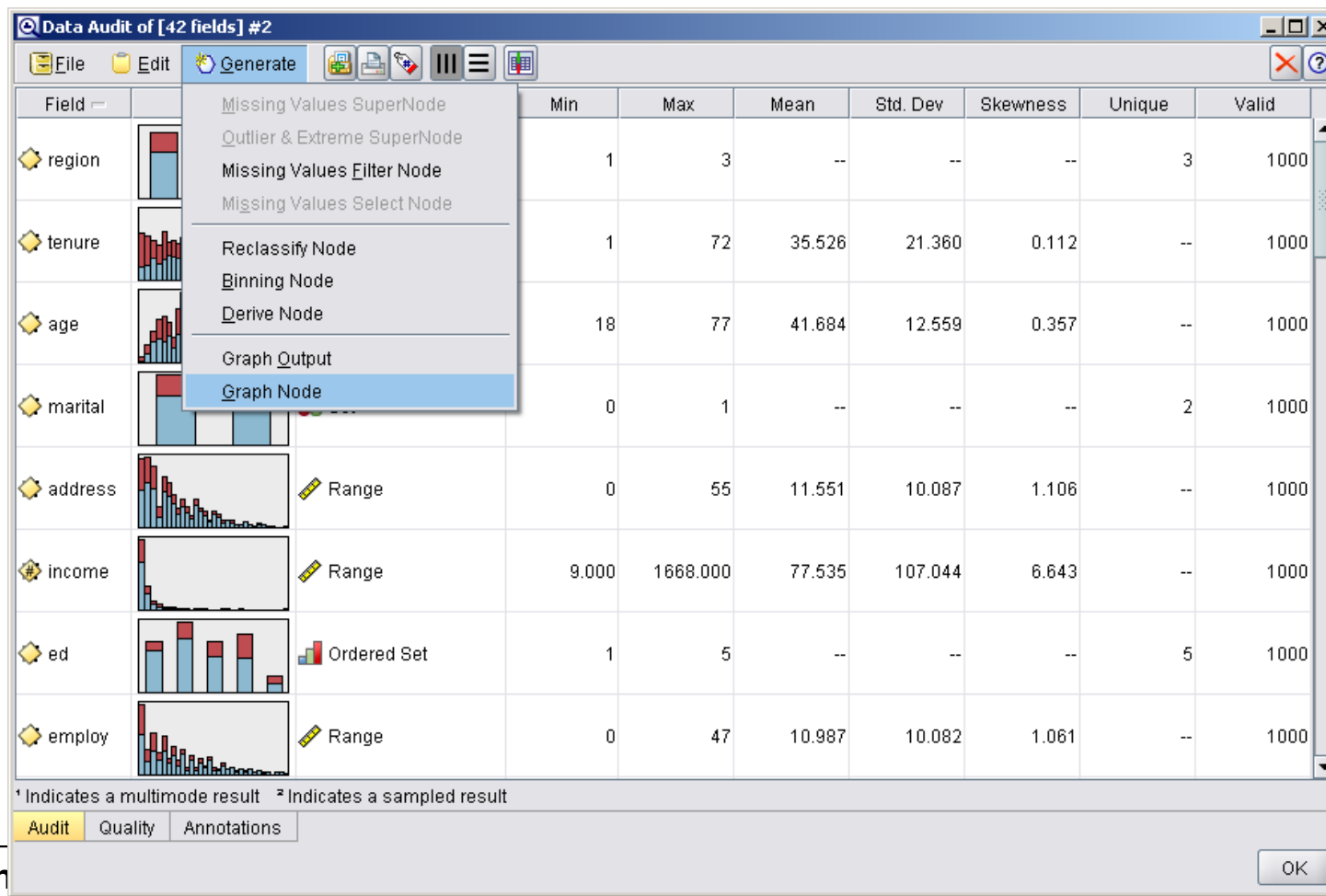
# Browsing Statistics and Charts

- Double-click on any thumbnail graph in the audit report to view a full-sized version of that chart. Because *churn* is the only target field in the stream, it is automatically used as an overlay.



# Browsing Statistics and Charts

- You can select one or more thumbnails and generate a **Graph** node for each. The generated nodes are placed on the stream canvas and can be added to the stream to re-create that particular graph.



---

# Handling Missing Values

# Handling Missing Values

- The **Quality** tab in the audit report displays information about outliers, extremes, and missing values.

**Data Audit of [42 fields] #2**

Complete fields (%): 90.48%    Complete records (%): 13.1%

Field	Type	Outliers	Extremes	Action	Impute Missing	Method	% C
region	Set	--	---		Never	Fixed	
tenure	Range	0	0 None		Never	Fixed	
age	Range	0	0 None		Never	Fixed	
marital	Set	--	---		Never	Fixed	
address	Range	12	0 None		Never	Fixed	
income	Range	9	6 None		Never	Fixed	
ed	Ordered Set	--	---		Never	Fixed	
employ	Range	8	0 None		Never	Fixed	
retire	Set	--	---		Never	Fixed	
gender	Set	--	---		Never	Fixed	
reside	Range	6	0 None		Never	Fixed	
tollfree	Set	--	---		Never	Fixed	
equip	Set	--	---		Never	Fixed	
callcard	Set	--	---		Never	Fixed	
wireless	Set	--	---		Never	Fixed	
longmon	Range	18	4 None		Never	Fixed	
tollmon	Range	9	1 None		Never	Fixed	
equipment	Range	2	0 None		Never	Fixed	
cardmon	Range	11	3 None		Never	Fixed	
wiremon	Range	8	1 None		Never	Fixed	
longten	Range	20	4 None		Never	Fixed	
tollten	Range	18	2 None		Never	Fixed	
equipten	Range	16	3 None		Never	Fixed	
cardten	Range	11	6 None		Never	Fixed	
wireten	Range	22	3 None		Never	Fixed	
multiline	Set	--	---		Never	Fixed	

Audit    **Quality**    Annotations

Clear    OK



# Handling Missing Values

- Quality tab

Data Audit of [42 fields] #2

Complete fields (%): 90.48% Complete records (%): 13.1%

Impute Missing	Method	% Complete	Valid Records	Null Value	Empty String	White Space	Blank Value
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	47.5	475	525	0	0	0
Never	Fixed	38.6	386	614	0	0	0
Never	Fixed	67.8	678	322	0	0	0
Never	Fixed	29.6	296	704	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0

Audit Quality Annotations

OK

# Missing Values

---

- Missing values are values in the dataset that are:
  - unknown,
  - uncollected, or
  - incorrectly entered.
- Usually, such values are invalid for their fields.
  - For example,
    - ◆ A value *Y* for the field *Sex* that should contain the values *M* and *F*.
    - ◆ A negative value for the field *Age* is meaningless and should also be interpreted as a blank.

# Handling Missing Values

---

- **Types of missing values in Clementine:**

- **Null values**

- ◆ These are **nonstring values** that have been left blank in the database or source file and have not been specifically defined as “missing” in a source or **Type** node.
    - ◆ Null values are displayed as System-missing **\$null\$**.
    - ◆ Note that empty strings are not considered nulls in Clementine, although they may be treated as nulls by certain databases.

- **Empty strings and white space**

- ◆ Empty string values and white space (strings with no visible characters) are treated as distinct from null values.
    - ◆ Empty strings are treated as equivalent to white space for most purposes.
    - ◆ For example, if you select the option to treat white space as blanks in a source or **Type** node, this setting applies to empty strings as well.

# Handling Missing Values

---

- **Types of missing values in Clementine:**

- **Reading in mixed data**

- ◆ Note that when you are reading in fields with numeric storage (either integer, real, time, timestamp, or date), any non-numeric values are set to null or system missing.

- **User-defined missing values**

- ◆ These are values such as unknown, 99, or -1 that are explicitly defined in a source node or **Type** node as missing.
    - ◆ Optionally, you can also choose to treat nulls and white space as blanks, which allows them to be flagged for special treatment and to be excluded from most calculations.

# Declare Missing Values

- To declare missing values or blanks
  - Double-clicking a field in the **Type** node opens a **Values Dialog Box**
  - Select **Define blanks** to activate the controls below that enable you to declare missing values or blanks in your data.

The screenshot shows the 'logtoll Values' dialog box. The 'Type' is set to 'Range' and 'Storage' is 'Real'. Under 'Values', 'Specify values and labels' is selected, with 'Lower' at 1.749199854809259 and 'Upper' at 5.153291594497779. The 'Define blanks' checkbox is checked, and a 'Missing values' list is visible. At the bottom, 'Null' is checked, and 'White space' is unchecked. The description is 'Log-toll free'.

# Declare Missing Values

---

- **Define blanks Options**

- **Missing values table**

- ◆ Allows you to define specific values (such as 99 or 0) as blanks.
    - ◆ The value should be appropriate for the storage type of the field

- **Range**

- ◆ Used to specify a range of missing values, for example, ages 1–17 or greater than 65.

- **White space**

- ◆ You can also specify white space (string values with no visible characters) as blanks.

# Handling Missing Values

---

- You should decide how to treat missing values in light of your business or domain knowledge.
  - In order to ease training time and increase accuracy, you may want to remove blanks from your dataset.
  - On the other hand, the presence of blank values may lead to new business opportunities or additional insights.
- In choosing the best technique, you should consider the following aspects of your data:
  - Size of the dataset
  - Number of fields containing blanks
  - Amount of missing information

# Handling Missing Values

---

- Two approaches to treat missing values:
  - You can exclude fields or records with missing values
  - You can impute, replace, or coerce missing values using a variety of methods



# Handling Records with Missing Values

---

- If the majority of missing values is concentrated in a small number of records, you can just exclude those records.
- Example,
  - a bank usually keeps detailed and complete records on its loan customers.
  - If, however, the bank is less restrictive in approving loans for its own staff members, data gathered for staff loans are likely to have several blank fields.
  - In such a case, there are two options for handling these missing values:
    - ◆ You can use a Select node to remove the staff records.
    - ◆ If the dataset is large, you can discard all records with blanks.

# Handling Records with Missing Values

- From the **Data Audit** browser, you can create a new **Select** node based on the results of the quality analysis.

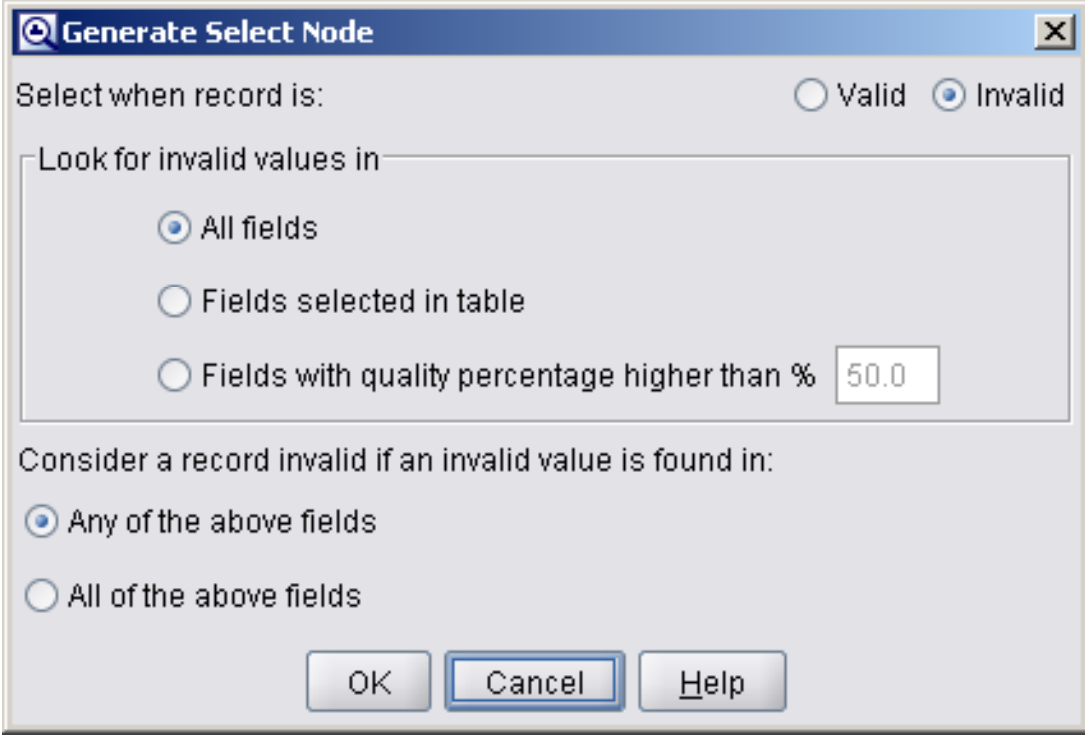
The screenshot shows the 'Data Audit of [42 fields] #1' window. The 'Generate' menu is open, and the 'Missing Values Select Node' option is highlighted. The window displays a table of quality analysis results for 42 fields. The table has columns for 'Field', 'Extremes', 'Action', 'Impute Missing', 'Method', and '% C'. The 'logtoll' field is highlighted in blue.

Field	Extremes	Action	Impute Missing	Method	% C
tollmon	1 None	Never	Fixed		
equipmon	0 None	Never	Fixed		
cardmon	3 None	Never	Fixed		
wiremon	1 None	Never	Fixed		
longten	4 None	Never	Fixed		
tollten	2 None	Never	Fixed		
equipten	3 None	Never	Fixed		
cardten	6 None	Never	Fixed		
wireten	3 None	Never	Fixed		
multline	---	Never	Fixed		
voice	---	Never	Fixed		
pager	---	Never	Fixed		
internet	---	Never	Fixed		
callid	---	Never	Fixed		
callwait	---	Never	Fixed		
forward	---	Never	Fixed		
confer	---	Never	Fixed		
ebill	---	Never	Fixed		
loglong	4	0 None	Never	Fixed	
logtoll	2	0 None	Never	Fixed	
logequi	1	0 None	Never	Fixed	
logcard	2	0 None	Never	Fixed	
logwire	1	0 None	Never	Fixed	
lninc	9	0 None	Never	Fixed	
custcat	--	---	Never	Fixed	
churn	--	---	Never	Fixed	

The 'logtoll' field is highlighted in blue. The 'Generate' menu is open, and the 'Missing Values Select Node' option is highlighted. The window also shows tabs for 'Audit', 'Quality', and 'Annotations' at the bottom.

# Handling Records with Missing Values

- **Generate Select node** dialog box



The image shows a screenshot of the 'Generate Select Node' dialog box in a software application. The dialog box has a title bar with a refresh icon and the text 'Generate Select Node'. It contains several options for selecting records based on missing values.

Select when record is: ☐ Valid ☒ Invalid

Look for invalid values in:

- ☒ All fields
- ☐ Fields selected in table
- ☐ Fields with quality percentage higher than %

Consider a record invalid if an invalid value is found in:

- ☒ Any of the above fields
- ☐ All of the above fields

At the bottom, there are three buttons: 'OK', 'Cancel', and 'Help'.

# Handling Records with Missing Values

---

- Generate Select node options:
  - Select when record is.
    - ◆ Specify whether records should be kept when they are Valid or Invalid.
  - Look for invalid values in.
    - ◆ Specify where to check for invalid values.
    - ◆ All fields.
      - The **Select** node will check all fields for invalid values.
    - ◆ Fields selected in table.
      - The Select node will check only the fields currently selected in the Quality output table.
    - ◆ Fields with quality percentage higher than.
      - The **Select** node will check fields where the percentage of complete records is greater than the specified threshold. The default threshold is 50%.

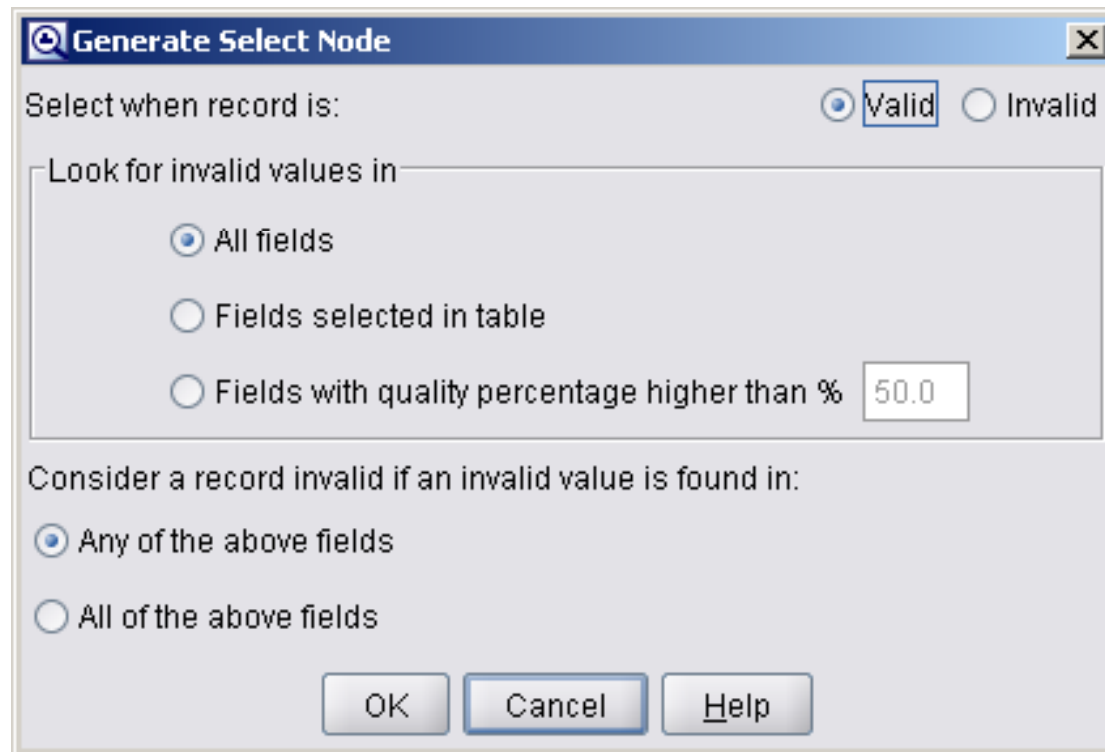
# Handling Records with Missing Values

---

- Consider a record invalid if an invalid value is found in.
  - ◆ Specify the condition for identifying a record as invalid.
  - ◆ Any of the above fields.
    - The **Select** node will consider a record invalid if any of the fields specified above contains an invalid value for that record.
  - ◆ All of the above fields.
    - The **Select** node will consider a record invalid only if all of the fields specified above contain invalid values for that record.

# Handling Records with Missing Values

- Select **Valid** option



# Handling Records with Missing Values

- The result

Data Audit of [42 fields] #2

Complete fields (%): 100% Complete records (%): 100%

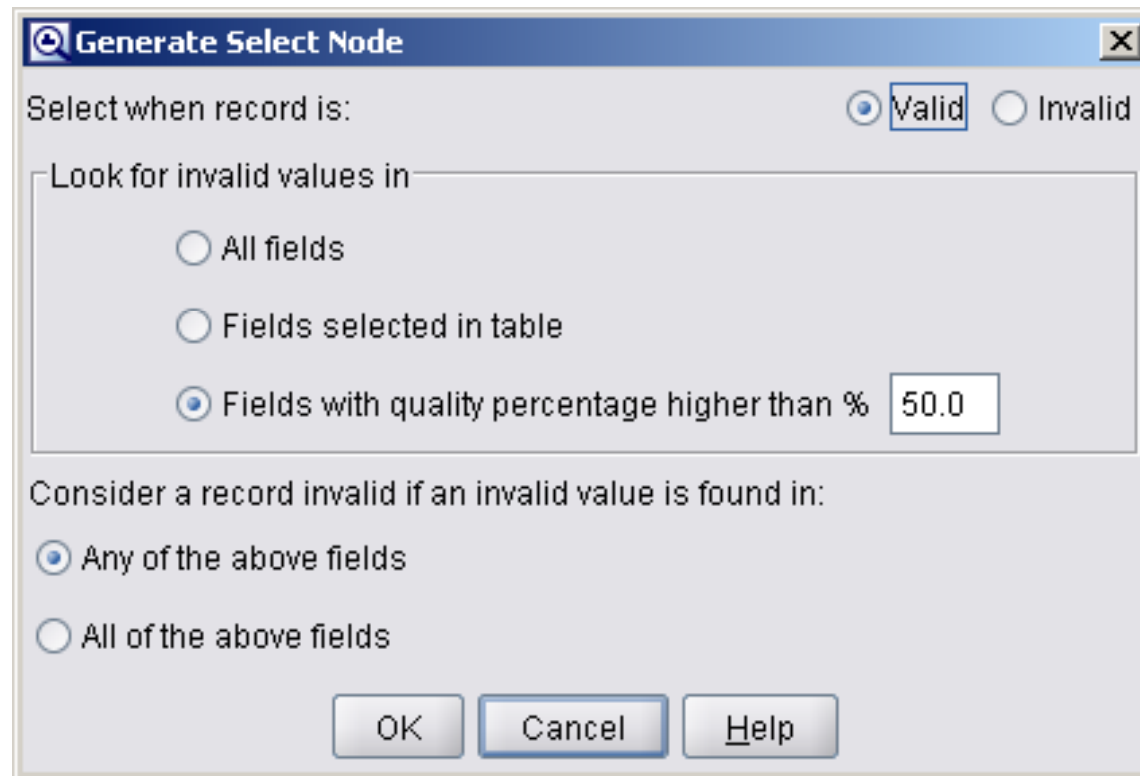
Action	Impute Missing	Method	% Complete	Valid Records	Null Value	Empty String	White Space
	Never	Fixed	100	131	0	0	0
	Never	Fixed	100	131	0	0	0
	Never	Fixed	100	131	0	0	0
	Never	Fixed	100	131	0	0	0
	Never	Fixed	100	131	0	0	0
	Never	Fixed	100	131	0	0	0
	Never	Fixed	100	131	0	0	0
	Never	Fixed	100	131	0	0	0
	Never	Fixed	100	131	0	0	0
	Never	Fixed	100	131	0	0	0
	Never	Fixed	100	131	0	0	0
	Never	Fixed	100	131	0	0	0
	Never	Fixed	100	131	0	0	0
	Never	Fixed	100	131	0	0	0
	Never	Fixed	100	131	0	0	0
	Never	Fixed	100	131	0	0	0
	Never	Fixed	100	131	0	0	0
	Never	Fixed	100	131	0	0	0
	Never	Fixed	100	131	0	0	0
	Never	Fixed	100	131	0	0	0
	Never	Fixed	100	131	0	0	0

Audit Quality Annotations

OK

# Handling Records with Missing Values

- Fields with quality percentage higher than.



The image shows a dialog box titled "Generate Select Node" with a close button (X) in the top right corner. The dialog is used to configure record selection based on missing values.

Select when record is: ☒ Valid ☐ Invalid

Look for invalid values in:

- ☐ All fields
- ☐ Fields selected in table
- ☒ Fields with quality percentage higher than %

Consider a record invalid if an invalid value is found in:

- ☒ Any of the above fields
- ☐ All of the above fields

Buttons: OK, Cancel, Help



# Handling Records with Missing Values

- Fields with quality percentage higher than.

Data Audit of [42 fields] #9

Complete fields (%): 92.86% Complete records (%): 19.32%

Extremes	Action	Impute Missing	Method	% Complete	Valid Records	Null Value	Empty S
1 None		Never	Fixed	100	678	0	
0 None		Never	Fixed	100	678	0	
3 None		Never	Fixed	100	678	0	
0 None		Never	Fixed	100	678	0	
4 None		Never	Fixed	100	678	0	
1 None		Never	Fixed	100	678	0	
0 None		Never	Fixed	100	678	0	
3 None		Never	Fixed	100	678	0	
2 None		Never	Fixed	100	678	0	
--		Never	Fixed	100	678	0	
--		Never	Fixed	100	678	0	
--		Never	Fixed	100	678	0	
--		Never	Fixed	100	678	0	
--		Never	Fixed	100	678	0	
--		Never	Fixed	100	678	0	
--		Never	Fixed	100	678	0	
--		Never	Fixed	100	678	0	
0 None		Never	Fixed	100	678	0	
0 None		Never	Fixed	60.029	407	271	
0 None		Never	Fixed	35.841	243	435	
0 None		Never	Fixed	100	678	0	
0 None		Never	Fixed	37.758	256	422	
0 None		Never	Fixed	100	678	0	
--		Never	Fixed	100	678	0	
--		Never	Fixed	100	678	0	

Audit Quality Annotations

OK

# Handling Fields with Missing Values

---

- If the majority of missing values is concentrated in a small number of fields, you can address them at the field level rather than at the record level.
  - For example, a market research company may collect data from a general questionnaire containing 50 questions. One of the questions address **age**, information that many people are reluctant to give. In this case, *age* have many missing values.
- This approach also allows you to experiment with the relative importance of particular fields before deciding on an approach for handling missing values.

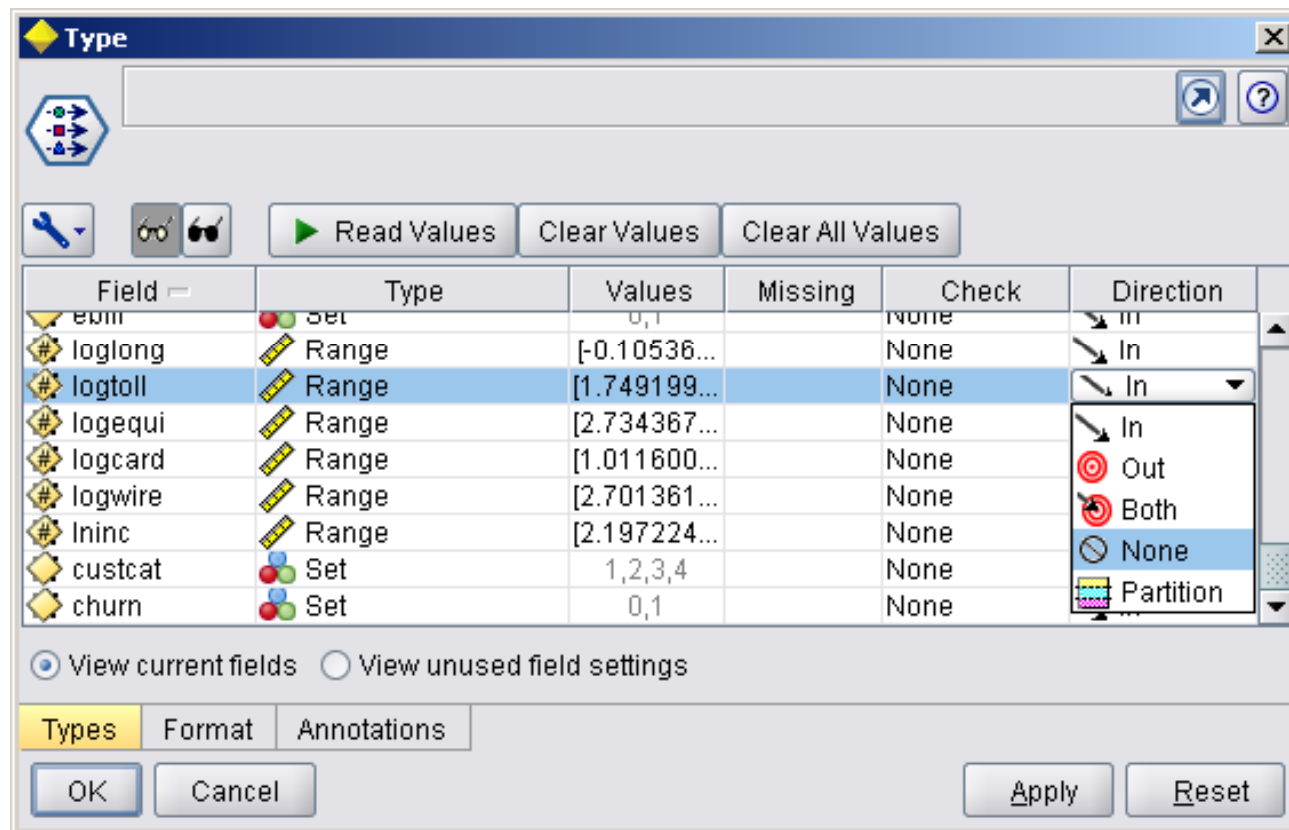
# Handling Fields with Missing Values

---

- **Options To Handle Fields with Missing Values:**
  - Using a **Type** node to set the fields' direction to **None**
    - ◆ This will keep the fields in the dataset but exclude them from the modeling processes.
  - Using a **Type** node to set the fields' direction to **None**
  - Filtering fields with missing data by using a **Data Audit** node to filter fields based on quality.
  - You can use a **Feature Selection** node to screen out fields with more than a specified percentage of missing values and to rank fields based on importance relative to a specified target.

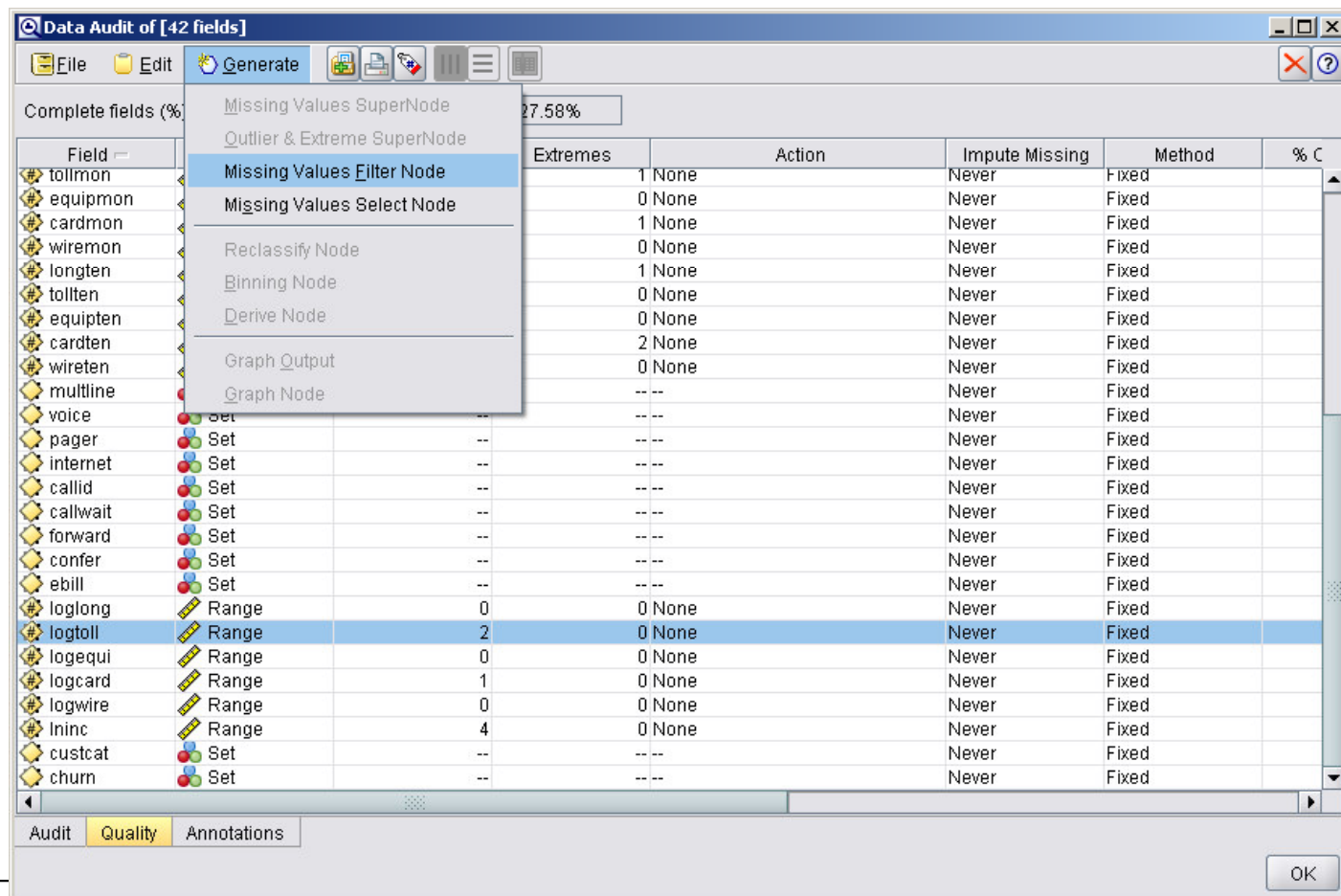
# Handling Fields with Missing Values

- Using a **Type** node to set the fields' direction to **None**



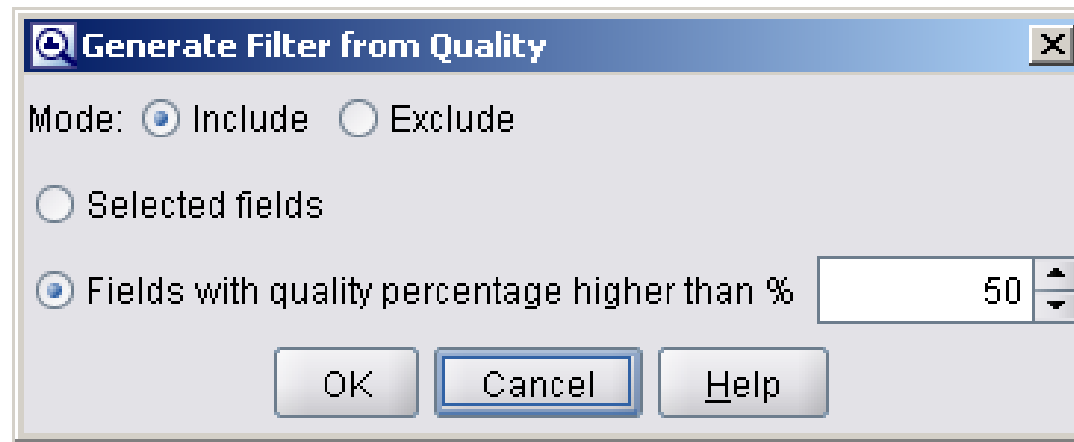
# Handling Fields with Missing Values

- From the **Data Audit** browser, you can create a new Filter node based on the results of the **Quality** analysis.



# Handling Fields with Missing Values

- **Generate Filter from Quality** dialog box



- **Mode**

- ◆ Select the desired operation for specified fields, either Include or Exclude.

# Handling Fields with Missing Values

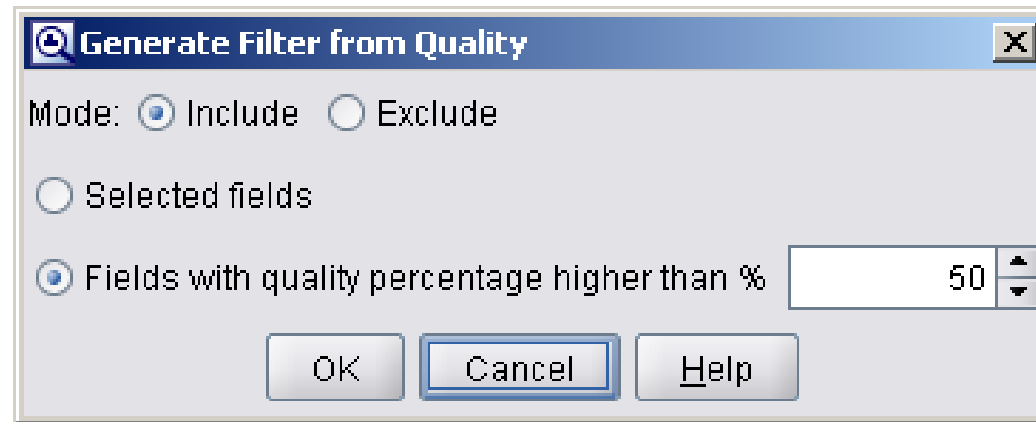
---

- **Generate Filter from Quality** dialog box options:
  - **Selected fields**
    - ◆ The **Filter** node will include/exclude the fields selected on the **Quality** tab.
    - ◆ For example you could sort the table on the % Complete column, use Shift-click to select the least complete fields, and then generate a Filter node that excludes these fields.
  - **Fields with quality percentage higher than**
    - ◆ The **Filter** node will include/exclude fields where the percentage of complete records is greater than the specified threshold.
    - ◆ The default threshold is 50%.

# Handling Fields with Missing Values

---

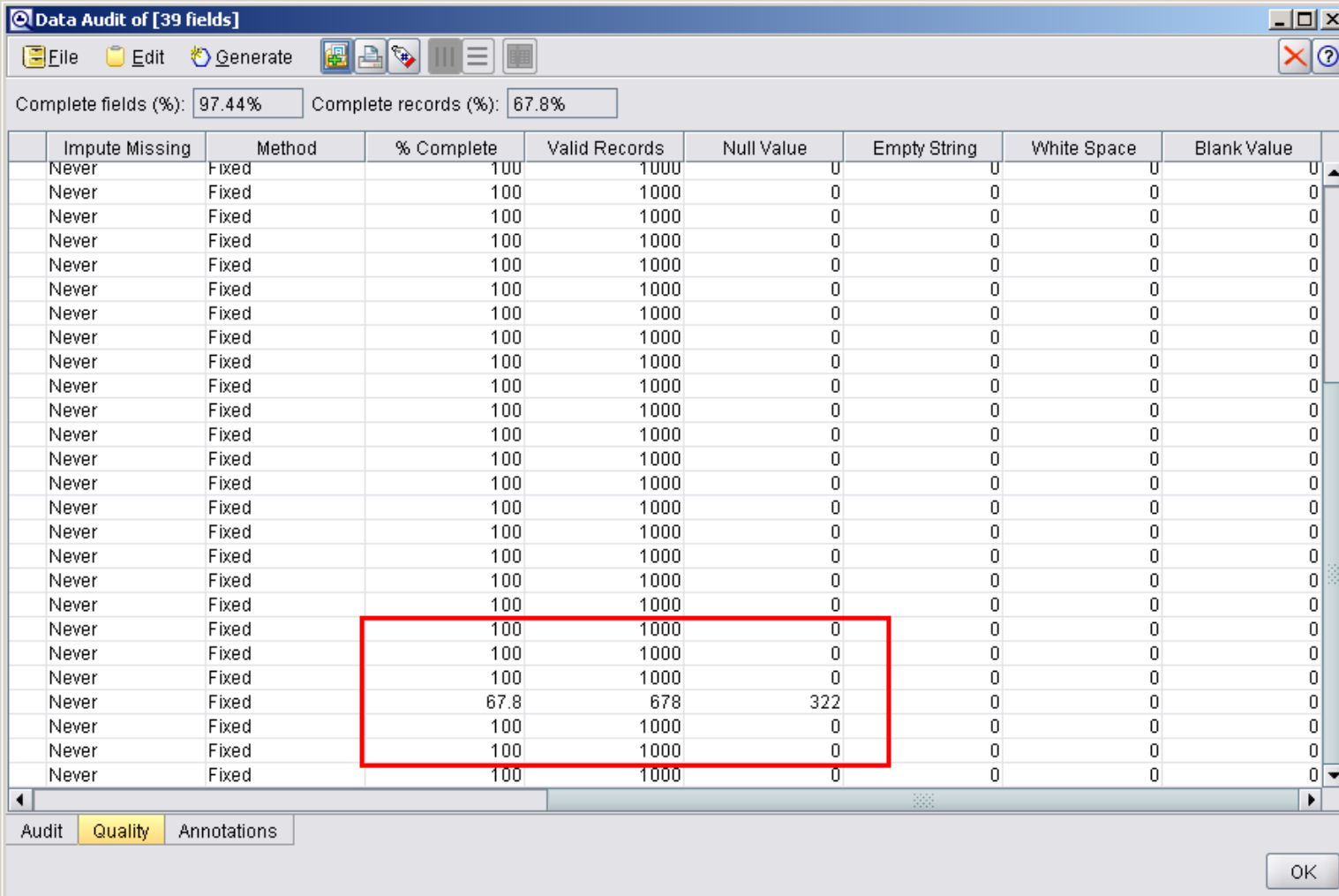
- Fields with quality percentage higher than 50 %





# Handling Fields with Missing Values

- Fields with quality percentage higher than 50 %



The screenshot shows the 'Data Audit of [39 fields]' window. At the top, it displays 'Complete fields (%)' as 97.44% and 'Complete records (%)' as 67.8%. Below this is a table with the following columns: Impute Missing, Method, % Complete, Valid Records, Null Value, Empty String, White Space, and Blank Value. The table lists 28 rows of data. A red box highlights the 25th row, which has a '% Complete' of 67.8, 'Valid Records' of 678, and 'Null Value' of 322. The bottom of the window has tabs for 'Audit', 'Quality', and 'Annotations', with 'Quality' currently selected. An 'OK' button is in the bottom right corner.

Impute Missing	Method	% Complete	Valid Records	Null Value	Empty String	White Space	Blank Value
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	67.8	678	322	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0
Never	Fixed	100	1000	0	0	0	0

# Imputing or Filling Missing Values

---

- **Imputing or Filling Missing Values**

- In cases where there are only a few missing values, it may be useful to insert values to replace the blanks.
- You can do this from the **Data Audit** report, which allows you to specify options for specific fields as appropriate and then generate a **SuperNode** that imputes values using a number of methods.
- This is the most flexible method, and it also allows you to specify handling for large numbers of fields in a single node.

# Imputing or Filling Missing Values

---

- **The methods for imputing missing values:**
  - **Fixed**
    - ◆ Substitutes a fixed value (either the field mean, midpoint of the range, or a constant that you specify).
  - **Random**
    - ◆ Substitutes a random value based on a normal or uniform distribution.
  - **Expression**
    - ◆ Allows you to specify a custom expression. For example, you could replace values with a global variable created by the Set Globals node.
  - **Algorithm**
    - ◆ Substitutes a value predicted by a model based on the C&RT algorithm.

# Imputing or Filling Missing Values

---

- **Algorithm method**

- For each field imputed using this method, there will be a separate C&RT model, along with a Filler node that replaces blanks and nulls with the value predicted by the model.
- A Filter node is then used to remove the prediction fields generated by the model.

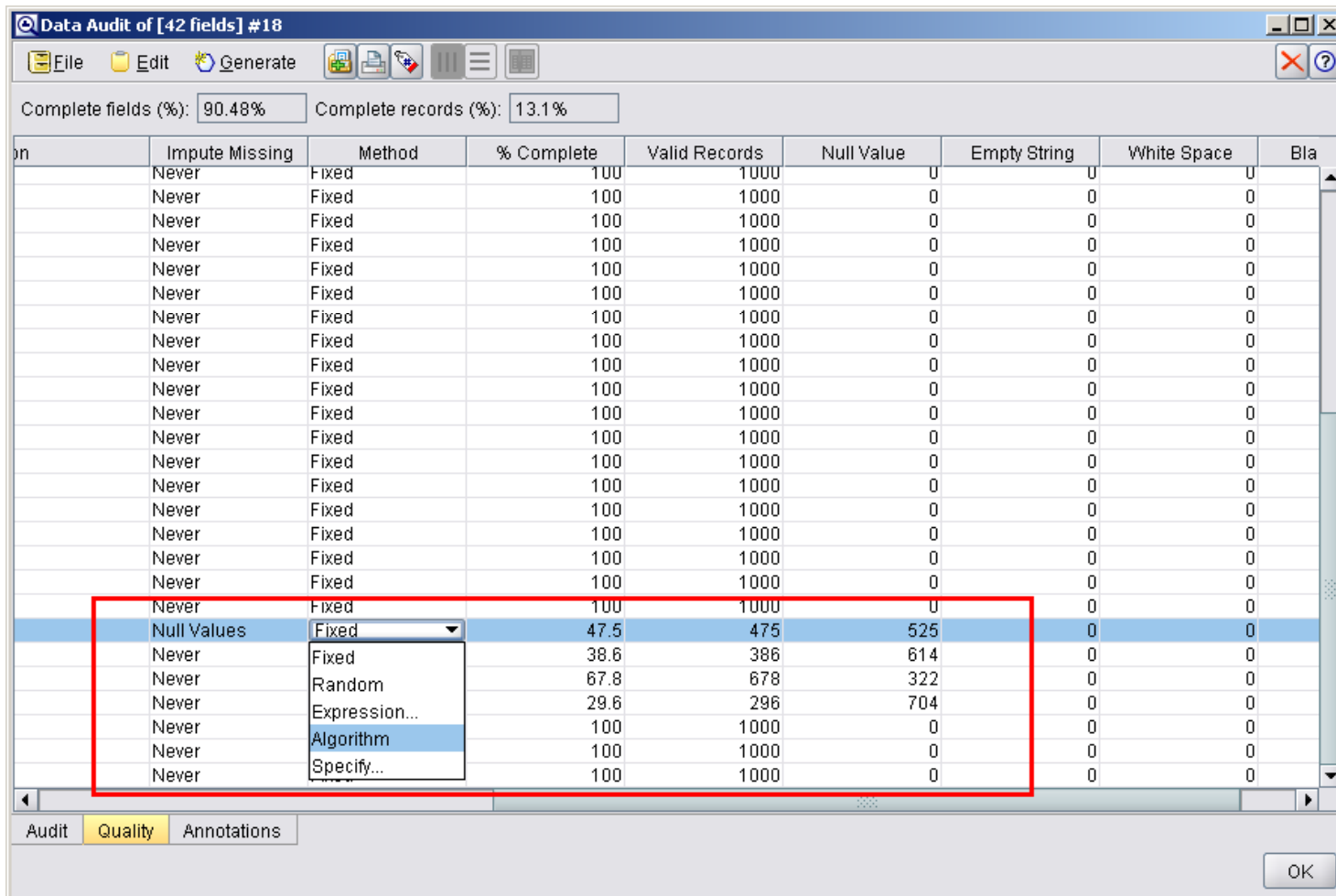
# Imputing or Filling Missing Values

---

- You can choose to impute missing values for specific fields as appropriate, and then generate a **SuperNode** to apply these transformations.
- In the **Impute Missing** column, specify the type of values you want to impute, if any.
- You can choose to impute **blanks**, **nulls**, **both**, or specify a **custom condition** or **expression** that selects the values to impute.

# Imputing or Filling Missing Values

- The algorithm method

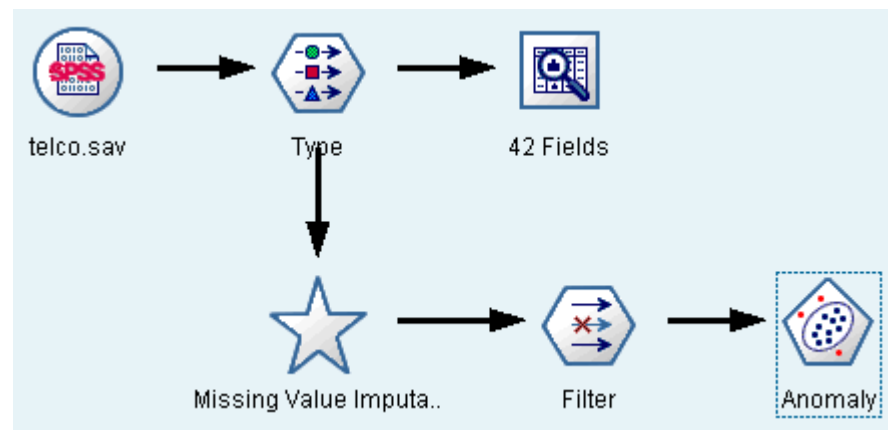
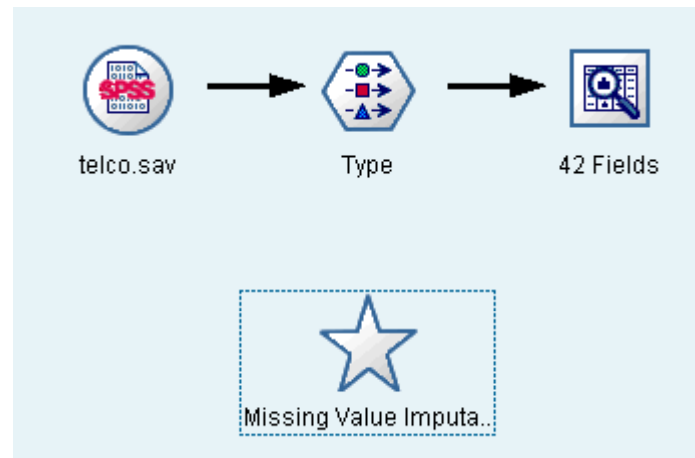


---



# Handling Outliers and Missing Values

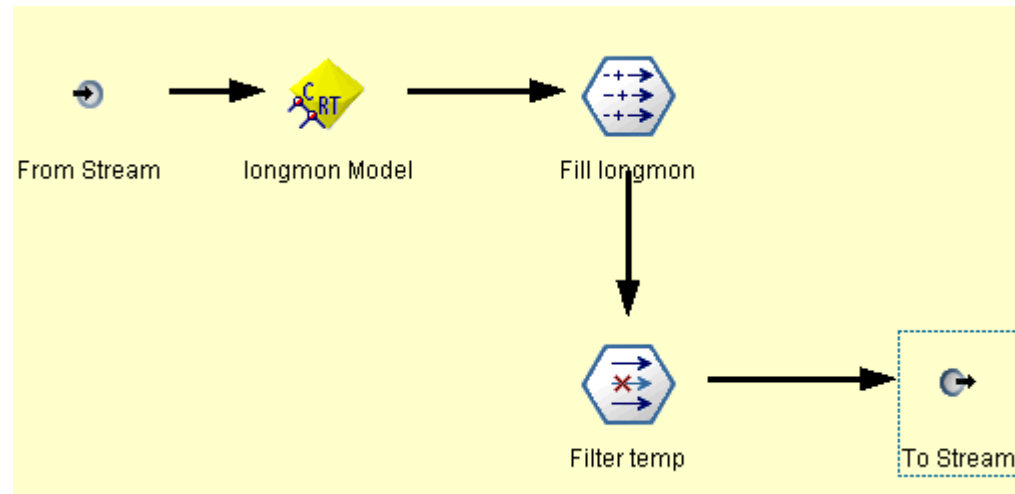
- The generated **SuperNode** is added to the stream canvas, where you can attach it to the stream to apply the transformations.





# Handling Outliers and Missing Values

- The **SuperNode** actually contains a series of nodes that perform the requested transformations.
- To understand how it works, you can edit the **SuperNode** and click **Zoom In**.



- For each field imputed using the algorithm method, for example, there will be a separate **C&RT** model, along with a Filler node that replaces blanks and nulls with the value predicted by the model. You can add, edit, or remove specific nodes within the **SuperNode** to further customize the behavior.

---

# Handling Outliers Values

# Handling Outliers Values

---

- The audit report lists number of outliers and extremes is listed for each field based on the detection options specified in the **Data Audit** node.
- You can choose to coerce, discard, or nullify these values for specific fields as appropriate, and then generate a **SuperNode** to apply the transformations.

# Handling Outliers Values

- The audit report lists number of outliers and extremes

Data Audit of [42 fields] #18

Complete fields (%): 90.48% Complete records (%): 13.1%

Field	Type	Outliers	Extremes	Action	Impute Missing	Method	% C
region	Set	--	--		Never	Fixed	
tenure	Range	0	0	None	Never	Fixed	
age	Range	0	0	None	Never	Fixed	
marital	Set	--	--		Never	Fixed	
address	Range	12	0	None	Never	Fixed	
income	Range	9	6	None	Never	Fixed	
ed	Ordered Set	--	--		Never	Fixed	
employ	Range	8	0	None	Never	Fixed	
retire	Set	--	--		Never	Fixed	
gender	Set	--	--		Never	Fixed	
reside	Range	6	0	None	Never	Fixed	
tollfree	Set	--	--		Never	Fixed	
equip	Set	--	--		Never	Fixed	
callcard	Set	--	--		Never	Fixed	
wireless	Set	--	--		Never	Fixed	
longmon	Range	18	4	None	Never	Fixed	
tollmon	Range	9	1	None	Never	Fixed	
equipmon	Range	2	0	None	Never	Fixed	
cardmon	Range	11	3	None	Never	Fixed	
wiremon	Range	8	1	None	Never	Fixed	
longten	Range	20	4	None	Never	Fixed	
tollten	Range	18	2	None	Never	Fixed	
equipten	Range	16	3	None	Never	Fixed	
cardten	Range	11	6	None	Never	Fixed	
wireten	Range	22	3	None	Never	Fixed	
multiline	Set	--	--		Never	Fixed	

Audit Quality Annotations

OK

# Handling Outliers Values

---

- In the **Action** column, specify handling for outliers and extremes for specific fields as desired.
- The actions are available for handling outliers and extremes:
  - **Coerce**
    - ◆ Replaces outliers and extreme values with the nearest value that would not be considered extreme.
    - ◆ For example if an outlier is defined to be anything above or below three standard deviations, then all outliers would be replaced with the highest or lowest value within this range.
  - **Discard**
    - ◆ Discards records with outlying or extreme values for the specified field.

# Handling Outliers Values

---

- Nullify
  - ◆ Replaces outliers and extremes with the null or system-missing value.
- Coerce outliers / discard extremes
  - ◆ Discards extreme values only.
- Coerce outliers / nullify extremes
  - ◆ Nullifies extreme values only.

# Handling Outliers Values

Data Audit of [42 fields] #22

File Edit Generate

Complete fields (%): 90.48% Complete records (%): 13.1%

Field	Type	Outliers	Extremes	Action	Impute Missing	Method	% C
region	Set	--	--	--	Never	Fixed	
tenure	Range	0	0	None	Never	Fixed	
age	Range	0	0	None	Never	Fixed	
marital	Set	--	--	--	Never	Fixed	
address	Range	12	0	None	Never	Fixed	
income	Range	9	6	None	Never	Fixed	
ed	Ordered Set	--	--	Coerce	Never	Fixed	
employ	Range	8	0	Discard	Never	Fixed	
retire	Set	--	--	Nullify	Never	Fixed	
gender	Set	--	--	Coerce outliers / discard extremes	Never	Fixed	
reside	Range	6	0	Coerce outliers / nullify extremes	Never	Fixed	
tollfree	Set	--	--	--	Never	Fixed	
equip	Set	--	--	--	Never	Fixed	
callcard	Set	--	--	--	Never	Fixed	
wireless	Set	--	--	--	Never	Fixed	
longmon	Range	18	4	None	Never	Fixed	
tollmon	Range	9	1	None	Never	Fixed	
equipmon	Range	2	0	None	Never	Fixed	
cardmon	Range	11	3	None	Never	Fixed	
wiremon	Range	8	1	None	Never	Fixed	
longten	Range	20	4	None	Never	Fixed	
tollten	Range	18	2	None	Never	Fixed	
equipten	Range	16	3	None	Never	Fixed	
cardten	Range	11	6	None	Never	Fixed	
wireten	Range	22	3	None	Never	Fixed	
multline	Set	--	--	--	Never	Fixed	

Audit Quality Annotations

OK

# Handling Outliers Values

Data Audit of [42 fields] #22

Complete fields (%) 3.1%

Field

Field	Type	Count	Extremes	Action	Impute Missing	Method	% C
region	Range	6	--		Never	Fixed	
tenure	Set	--	0 None		Never	Fixed	
age	Set	--	0 None		Never	Fixed	
marital	Set	--	--		Never	Fixed	
address	Set	--	0 Discard		Never	Fixed	
income	Set	--	6 None		Never	Fixed	
ed	Set	--	--		Never	Fixed	
employ	Set	--	--		Never	Fixed	
retire	Set	--	--		Never	Fixed	
gender	Set	--	--		Never	Fixed	
reside	Set	--	--		Never	Fixed	
tollfree	Set	--	--		Never	Fixed	
equip	Set	--	--		Never	Fixed	
callcard	Set	--	--		Never	Fixed	
wireless	Set	--	--		Never	Fixed	
longmon	Range	18	--		Never	Fixed	
tollmon	Range	9	--		Never	Fixed	
equipmon	Range	2	0 None		Never	Fixed	
cardmon	Range	11	3 None		Never	Fixed	
wiremon	Range	8	1 None		Never	Fixed	
longten	Range	20	4 None		Never	Fixed	
tollten	Range	18	2 None		Never	Fixed	
equipten	Range	16	3 None		Never	Fixed	
cardten	Range	11	6 None		Never	Fixed	
wireten	Range	22	3 None		Never	Fixed	
multline	Set	--	--		Never	Fixed	

Outlier SuperNode

Generate SuperNode for:

☐ All fields ☒ Selected fields only

OK Cancel



# Handling Outliers Values

---

- After completing the audit and adding the generated nodes to the stream, you can proceed with your analysis.
- Optionally, you may want to further screen your data using **Anomaly Detection**, **Feature Selection**, or a number of other methods.

---

# References

# References

---

- Integral Solutions Limited., **Clementine® 12.0 Applications Guide**, 2007. (chapter 7)



The end