

---

# **Data Mining**

## **SPSS Clementine 12.0**

### **7. Two-Step Clustering Algorithm**

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

# Outline

---

- **TwoStep Algorithm in Clementine**
- **TwoStep Node**
- **References**

---

# **TwoStep Algorithm in Clementine**

# Overview

---

- The **TwoStep cluster method** is a scalable cluster analysis algorithm designed to handle very large data sets.
- The SPSS TwoStep Cluster Component:
  - Handles both continuous and categorical variables by extending the model-based distance measure
  - Utilizes a two-step clustering approach similar to BIRCH (Zhang et al. 1996)
  - Provides the capability to automatically find the optimal number of clusters

# Overview

---

- It has two steps:
  - 1) **Pre-clustering Step**: pre-cluster the cases (or records) into many small sub-clusters;
  - 2) **Clustering Step**: cluster the sub-clusters resulting from pre-cluster step into the desired number of clusters.

# Pre-clustering Step

---

- The pre-cluster step uses a sequential clustering approach.
- It scans the data records one by one and decides if the current record should be merged with the previously formed clusters or starts a new cluster based on the distance criterion.
- The procedure is implemented by constructing a modified **cluster feature (CF) tree** (like **BIRCH algorithm**)

# Pre-clustering Step

---

- The CF tree consists of levels of nodes, and each node contains a number of entries.
- A **leaf entry** (an entry in the leaf node) represents a final sub-cluster.
- The non-leaf nodes and their entries are used to guide a new record quickly into a correct leaf node.
- Each entry is characterized by its CF that consists:
  - the entry's number of records
  - mean and variance of each range field
  - counts for each category of each symbolic field

# Pre-clustering Step

---

- For each successive record, starting from the root node, it is recursively guided by the closest entry in the node to find the closest child node, and descends along the CF tree.
- Upon reaching a leaf node, it finds the closest leaf entry in the leaf node.
- If the record is within a threshold distance of the closest leaf entry, it is absorbed into the leaf entry and the CF of that leaf entry is updated.
- Otherwise it starts its own leaf entry in the leaf node.



# Pre-clustering Step

---

- If there is no space in the leaf node to create a new leaf entry, the leaf node is split into two.
- The entries in the original leaf node are divided into two groups using the farthest pair as seeds, and redistributing the remaining entries based on the closeness criterion.
- If the CF tree grows beyond allowed maximum size, the CF tree is rebuilt based on the existing CF tree by increasing the threshold distance criterion.
- The rebuilt CF tree is smaller and hence has space for new input records.
- This process continues until a complete data pass is finished.

# Pre-clustering Step

---

- These properties of CF make it possible to maintain only the entry CFs, rather than the sets of individual records.
- Hence the CF-tree is much smaller than the original data and can be stored in memory more efficiently.

# Clustering Step

---

- The cluster step takes sub-clusters resulting from the pre-cluster step as input and then groups them into the desired number of clusters.
- Since the number of sub-clusters is much less than the number of original records, traditional clustering methods can be used effectively.
- TwoStep uses an **agglomerative hierarchical clustering method**, because it works well with the auto-cluster method.

# Clustering Step

---

- **Hierarchical clustering**
  - refers to a process by which clusters are recursively merged, until at the end of the process only one cluster remains containing all records.
- The process starts by defining a starting cluster for each of the sub-clusters produced in the pre-cluster step.
- All clusters are then compared, and the pair of clusters with the smallest distance between them is selected and merged into a single cluster.

# Clustering Step

---

- After merging, the new set of clusters is compared, the closest pair is merged, and the process repeats until all clusters have been merged.
- Because the clusters are merged recursively in this way, it is easy to compare solutions with different numbers of clusters.
- To get a five-cluster solution, simply stop merging when there are five clusters left; to get a four-cluster solution, take the five-cluster solution and perform one more merge operation, and so on.

# Number of Clusters (auto-clustering)

---

- TwoStep can use the hierarchical clustering method in the second step to assess multiple cluster solutions and automatically determine the **optimal number of clusters** for the input data.
- A hierarchical clustering produces a sequence of partitions in one run: 1, 2, 3, ... clusters.
- In contrast, a  $k$ -means algorithm would need to run multiple times (one for each specified number of clusters) in order to generate the sequence.

# Performance of the TwoStep Clustering

---

- SPSS implemented the TwoStep Cluster Component in both Java and C++ language.
- It tested the performance of the TwoStep Cluster Component on simulated datasets.
- Results are presented in the table below.
- The total time used for each dataset shown here comes from a Java implementation using text input files run on a Pentium 800Mhz, 256MB RAM computer.

# Performance of the TwoStep Clustering

- Performance of the TwoStep Clustering

Datasets	Number of records (x1000)	Number of variables		True number of clusters	Number of sub-clusters by pre-cluster	No. of clusters found by auto-cluster	Percentage of wrongly clustered	Total time used (in seconds)
		Con	Cat*					
Data 1	200	2	0	5	199	5	0.09%	21
Data 2	400	2	0	5	269	5	0.03%	41
Data 3	500	2	0	5	207	5	0.35%	47
Data 4	1,000	2	0	5	297	5	1.2%	93
Data 5	2,000	2	0	5	243	5	0.07%	193
Data 6	2,500	2	0	5	187	5	0.11%	229
Data 7	8.4	640	0	7	71	7	0%	177
Data 8	1,000	0	5	4	243	4	0.03%	572
Data 9	1,000	5	5	8	232	8	0%	1,070
Data 10	1,000	25	25	10	264	10	0%	4,970
*All the categorical variables are of 12 categories.								



---

# TwoStep Cluster Node

# TwoStep Cluster Node

---

- **Note:**

- The resulting model depends to a certain extent on the order of the training data.
- Reordering the data and rebuilding the model may lead to a different final cluster model.

- **Requirements.**

- you need one or more *In* fields.
- Fields with direction *Out*, *Both*, or *None* are ignored.

# TwoStep Cluster Node

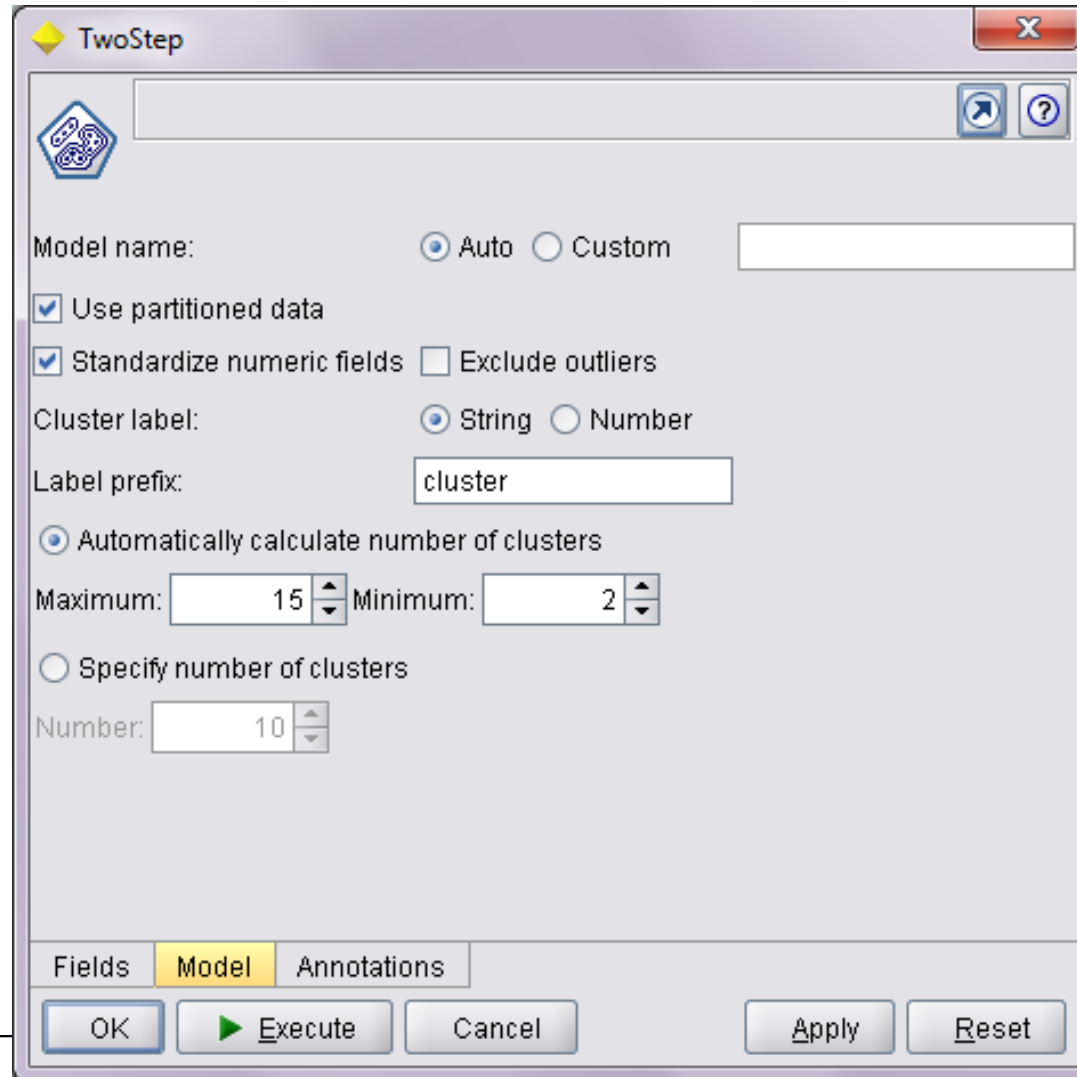
---

- **Missing values**

- The TwoStep Cluster algorithm does not handle missing values.
- Records with blanks for any of the input fields will be ignored when building the model.

# TwoStep Cluster Node Model Options

- This node is available with the Segmentation module.



The image shows the 'TwoStep' dialog box in the Clementine software. The dialog has a title bar with a yellow diamond icon and the text 'TwoStep'. Below the title bar is a toolbar with a blue icon and a help button. The main area contains the following options:

- Model name:** Radio buttons for 'Auto' (selected) and 'Custom'. A text field is next to the 'Custom' button.
- ☒ **Use partitioned data**
- ☒ **Standardize numeric fields** ☐ **Exclude outliers**
- Cluster label:** Radio buttons for 'String' (selected) and 'Number'.
- Label prefix:** A text field containing 'cluster'.
- ☒ **Automatically calculate number of clusters**
  - Maximum:** A spinner box set to 15.
  - Minimum:** A spinner box set to 2.
- ☐ **Specify number of clusters**
  - Number:** A spinner box set to 10.

At the bottom, there are three tabs: 'Fields', 'Model' (selected), and 'Annotations'. Below the tabs are five buttons: 'OK', 'Execute' (with a green play icon), 'Cancel', 'Apply', and 'Reset'.

# TwoStep Cluster Node Model Options

---

- **Model name.**

- You can generate the model name automatically based on the target or ID field or specify a custom name.

- **Use partitioned data.**

- If a partition field is defined, this option ensures that data from only the training partition is used to build the model.

- **Standardize numeric fields.**

- By default, TwoStep will standardize all numeric input fields to the same scale, with a mean of 0 and a variance of 1.
- To retain the original scaling for numeric fields, deselect this option.
- Symbolic fields are not affected.

# TwoStep Cluster Node Model Options

---

- **Exclude outliers.**

- If you select this option, records that don't appear to fit into a substantive cluster will be automatically excluded from the analysis.
- This prevents such cases from distorting the results.
- Outlier detection occurs during the preclustering step.
- When this option is selected, subclusters with few records relative to other subclusters are considered potential outliers, and the tree of subclusters is rebuilt excluding those records.

# TwoStep Cluster Node Model Options

---

- **Cluster label.**

- Specify the format for the generated cluster membership field.
- Cluster membership can be indicated as a String with the specified Label prefix (for example, "Cluster 1", "Cluster 2", and so on) or as a Number.

- **Automatically calculate number of clusters.**

- Specify a range of solutions to try by setting the Maximum and the Minimum number of clusters.
- The largest change in distance is used to identify the final cluster model.

# TwoStep Cluster Node Model Options

---

- **Specify number of clusters.**
  - If you know how many clusters to include in your model, select this option and enter the number of clusters.

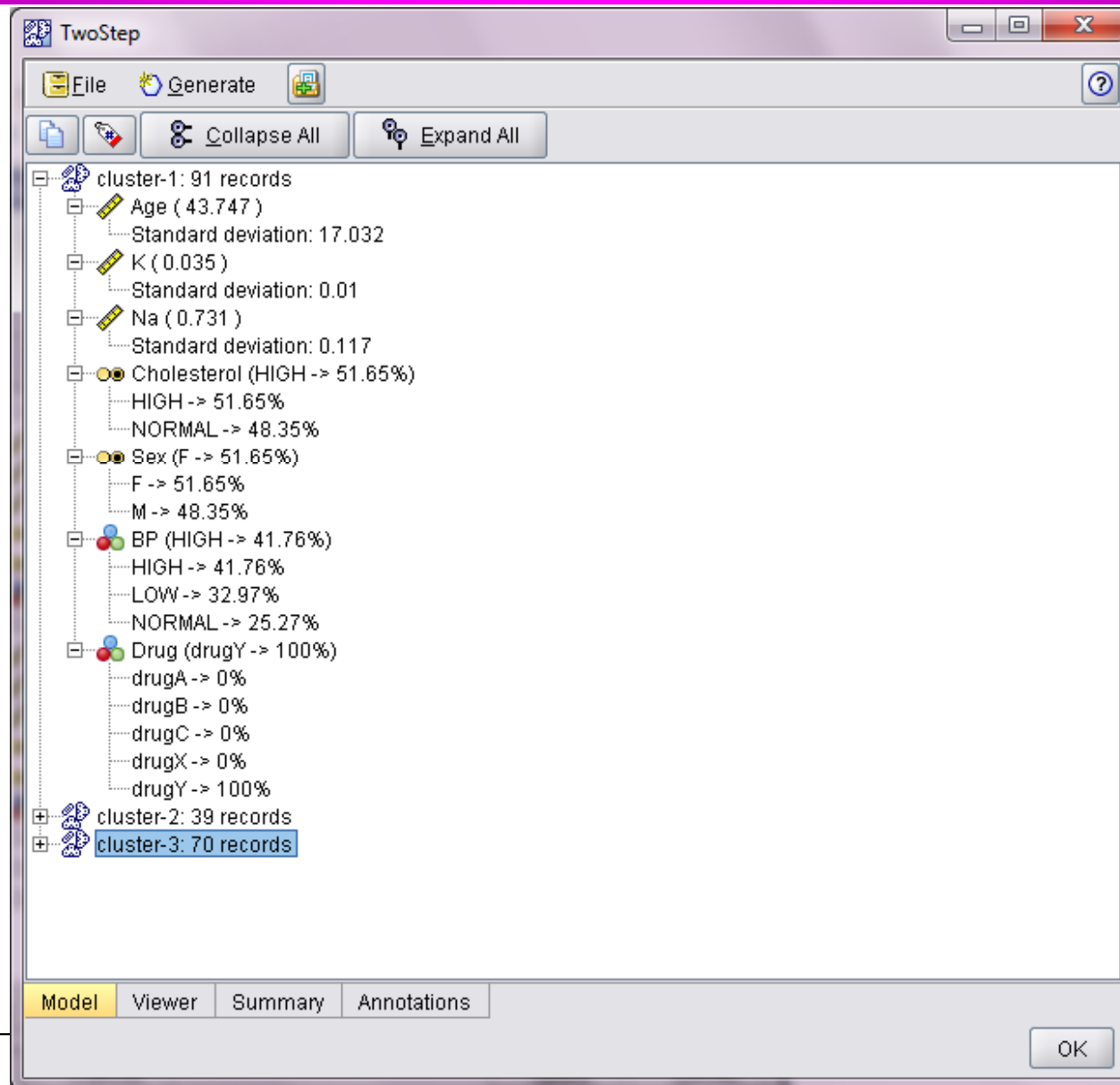


# TwoStep Cluster Model Nuggets

---

- When you execute a stream containing a TwoStep cluster model nugget, the node adds a new field containing the cluster membership for that record.
- The new field name is derived from the model name, prefixed by *\$T-*.
- For example, if your model is named TwoStep, the new field would be named *\$T-TwoStep*.

# TwoStep Model Cluster Details

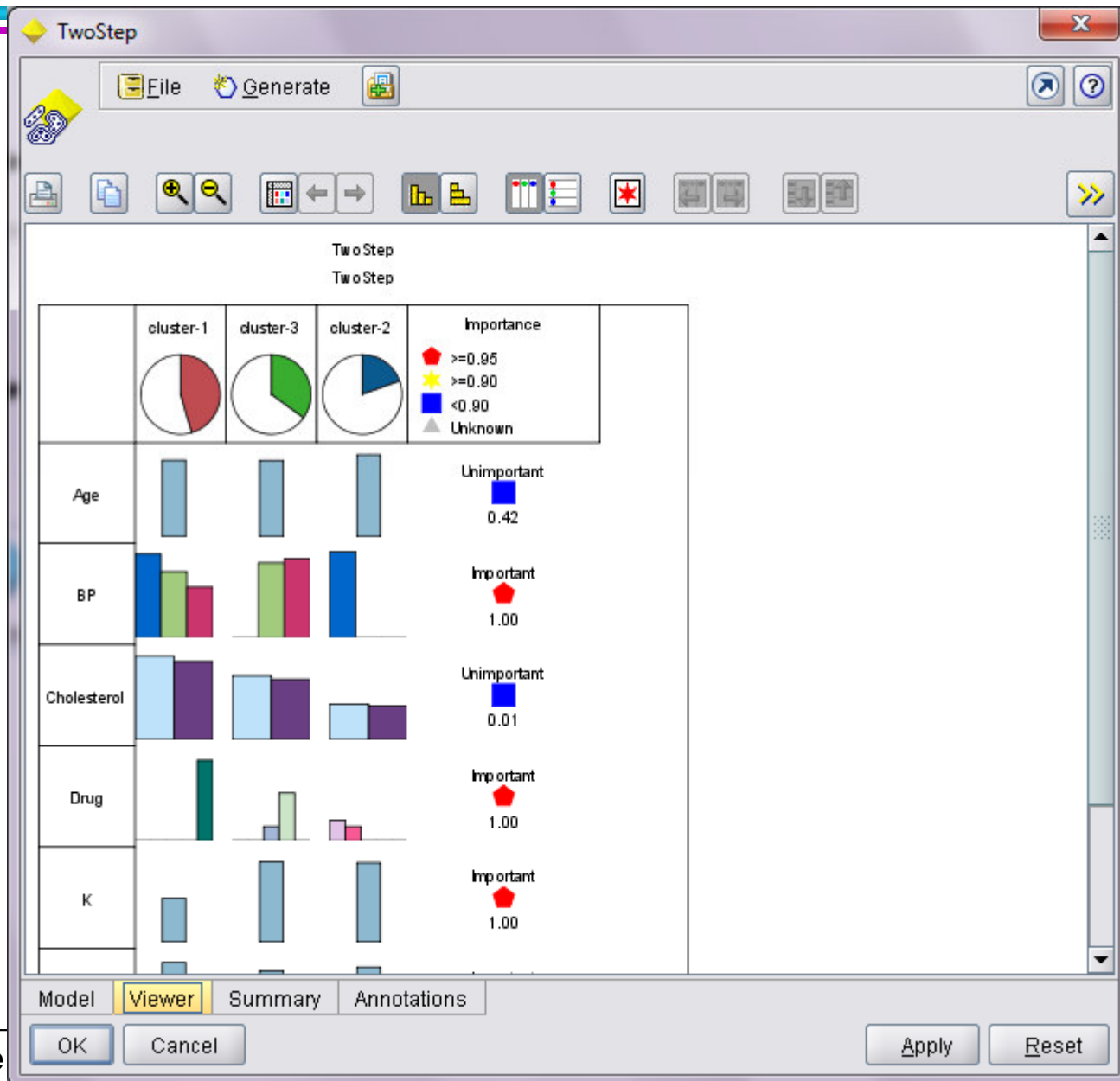


# TwoStep Model Cluster Details

---

- The **Model** tab for a TwoStep cluster model nugget contains detailed information about the clusters defined by the model.
- Clusters are labeled, and the number of records assigned to each cluster is shown.
- Each cluster is described by its center, which can be thought of as the **prototype** for the cluster.
- For scale fields, the average value and standard deviation for training records assigned to the cluster are given
- For symbolic fields, the proportion for each distinct value is reported.

# TwoStep Viewer Tab



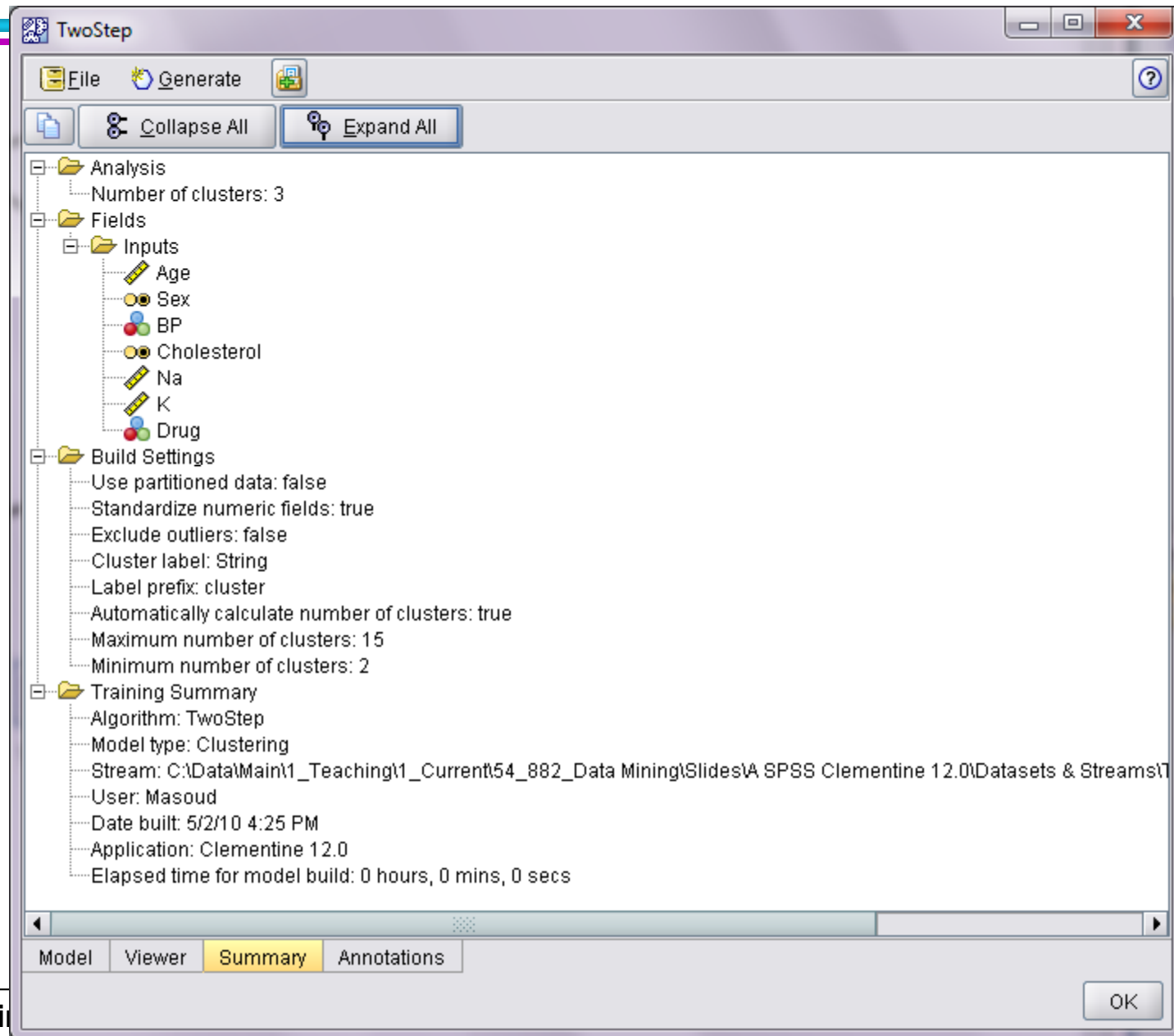
Clementine

# TwoStep Viewer Tab

---

- The **Viewer tab** shows a graphical display of summary statistics and distributions for fields between clusters.
- By default, the clusters are displayed on the  $x$  axis and the fields on the  $y$  axis.

# TwoStep Model Summary



# TwoStep Model Summary

---

- The **Summary** tab for a TwoStep cluster model nugget displays the number of clusters found, along with information about the training data, the estimation process, and build settings used.

---

# References



# References

---

- Integral Solutions Limited., **Clementine® 12.0 Algorithms Guide**, 2007. (Chapter 10)
- Integral Solutions Limited., **Clementine® 12.0 Modeling Nodes**, 2007. (Chapter 11)
- Zhang, T., R. Ramakrishnon, and M. Livny. 1996. **BIRCH: An efficient data clustering method for very large databases**. In: Proceedings of the ACM SIGMOD Conference on Management of Data, Montreal, Canada: ACM, 103–114, 1996.



The end