**Experiment 5**

**Aim:**

Experiment to explore Rapid Miner and implement classification models like Decision Tree and Naive Bayes etc.

**Theory:**

In machine learning, a decision tree is a predictive model that maps observations about an item to conclusions about its target value. It is a tree-structured model where each internal node represents a test on an attribute, each branch represents the outcome of the test, and each leaf node represents a class label. Decision trees can be used for both regression and classification tasks, and they are widely used in various applications, such as finance, medicine, and engineering. The goal of building a decision tree is to create a model that predicts the value of a target variable based on several input variables. The decision tree algorithm uses a recursive process to split the data into smaller subsets based on the input variables until it reaches a point where it can make a prediction.

The basic steps involved in creating a decision tree are as follows:

1. Collect and prepare data: This involves collecting and organizing the data, and then preparing it for analysis. This may include data cleaning, data transformation, and feature selection.
2. Choose an algorithm: There are various algorithms available for building decision trees, including ID3, C4.5, CART, and CHAID. The choice of algorithm will depend on the specific problem and the characteristics of the data.
3. Build the tree: This involves applying the chosen algorithm to the data to create the decision tree. The tree is built by recursively partitioning the data into subsets based on the values of the input features, and selecting the feature that provides the most information gain at each step.
4. Evaluate the tree: Once the tree is built, it needs to be evaluated to determine its accuracy and effectiveness. This may involve using cross-validation or other techniques to estimate the performance of the tree on new data.
5. Use the tree: Finally, the decision tree can be used to make predictions on new data. This involves traversing the tree from the root to a leaf node based on the values of the input features, and outputting the corresponding class label or value.
6. **Gini Index-**

Gini index is a measure of impurity or inequality used in decision trees to determine the purity of a given split in the data. It ranges from 0 to 1, where 0 represents a completely pure split (all observations belong to the same class) and 1 represents a completely impure split (an equal number of observations belong to each class).

The formula for calculating the Gini index is:

Gini = 1 - (p\_1)^2 - (p\_2)^2 - ... - (p\_k)^2

where k is the number of classes, and p\_i is the proportion of observations belonging to class i in the split.

1. **Information Gain-**

Information Gain is a measure used in decision tree algorithms to determine the relevance of a feature to a target variable. It measures the reduction in entropy or degree of disorder in the target variable when a feature is used to split the data into subsets. The formula for Information Gain is:

Information Gain = Entropy(parent) - [Weighted Avg. \* Entropy(children)]

where,

Entropy(parent) is the entropy of the target variable for the entire dataset

Weighted Avg. is the weighted average of the entropy for each child node

Entropy(children) is the entropy of the target variable for each child node

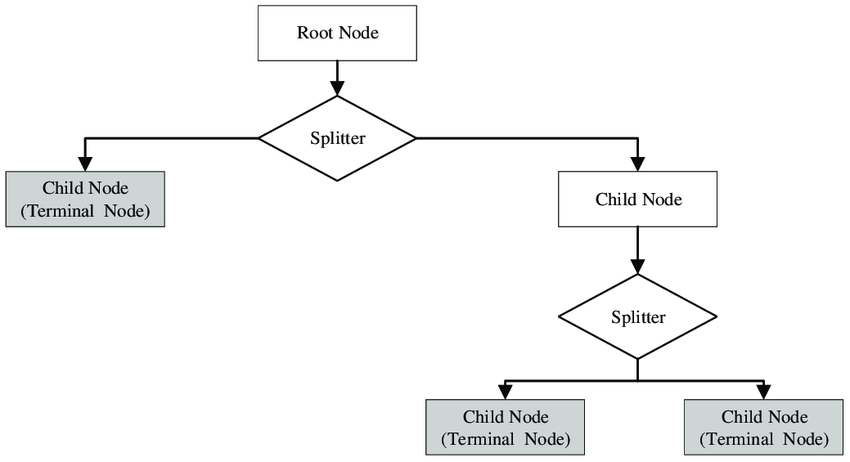
1. **Entropy**-

In the context of machine learning and decision trees, entropy is a measure of impurity or disorder within a set of examples. It is used to quantify the amount of uncertainty or randomness in a set of data. A dataset with low entropy is considered more uniform and has less randomness, whereas a dataset with high entropy is considered more disordered and has more randomness.

The entropy of a dataset S with respect to a binary target variable Y can be calculated using the following formula:

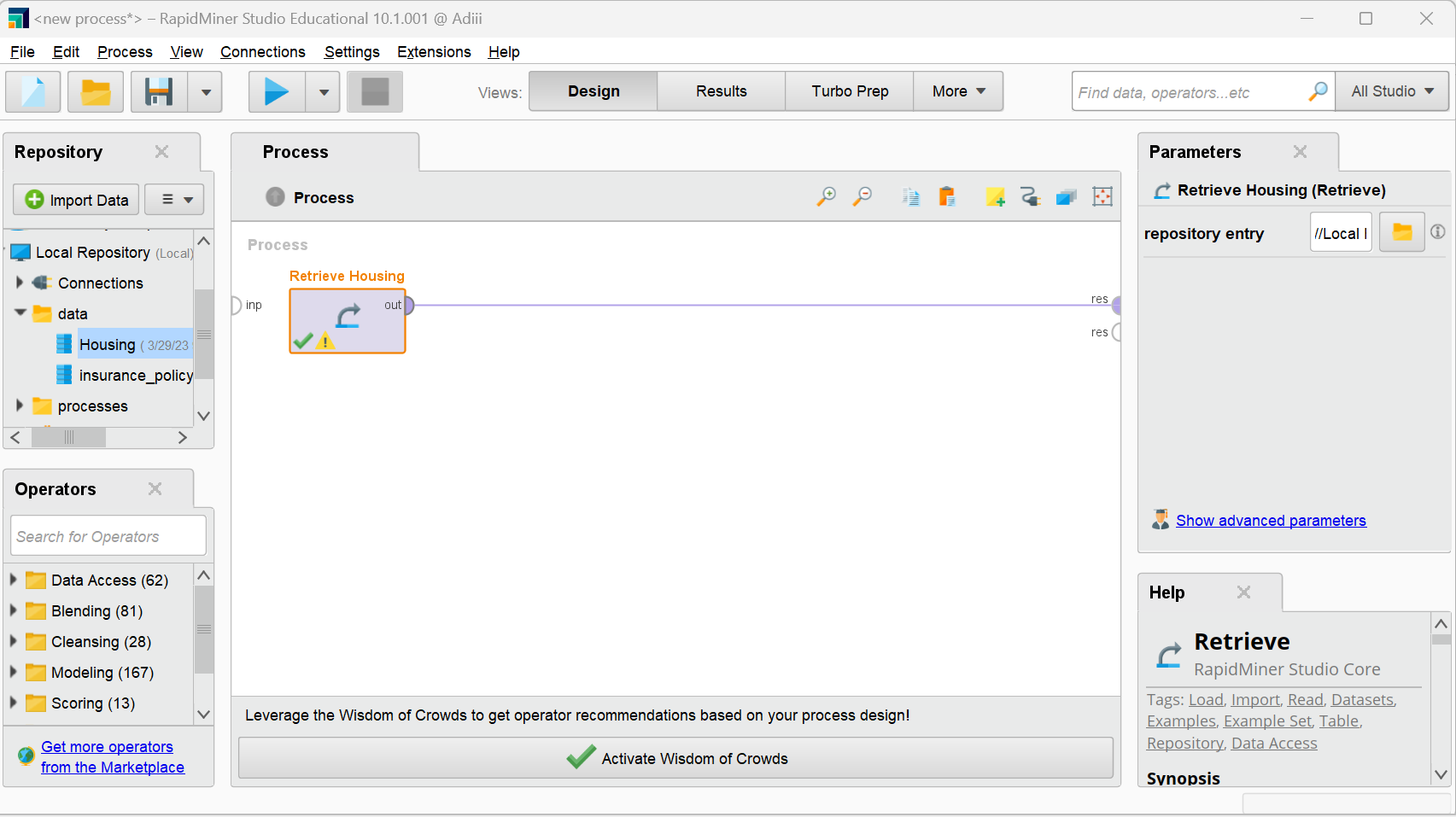
Entropy(S) = -p\_1 \log\_2 p\_1 - p\_2 \log\_2 p\_2$

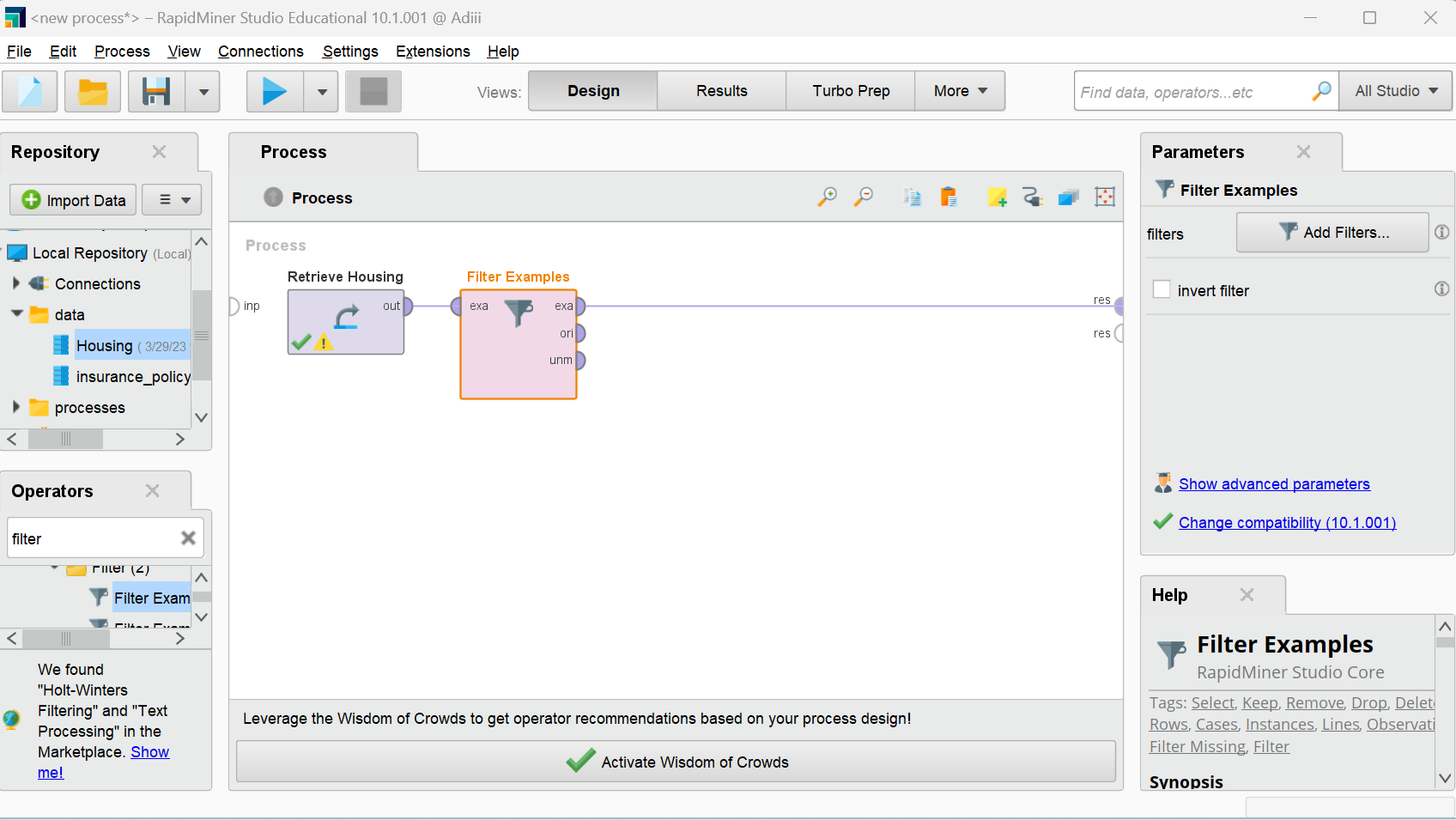
where p1 is the proportion of examples in S that belong to class 1, and p2 is the proportion of examples in S that belong to class 2. The logarithm base 2 is used to measure the entropy in bits.

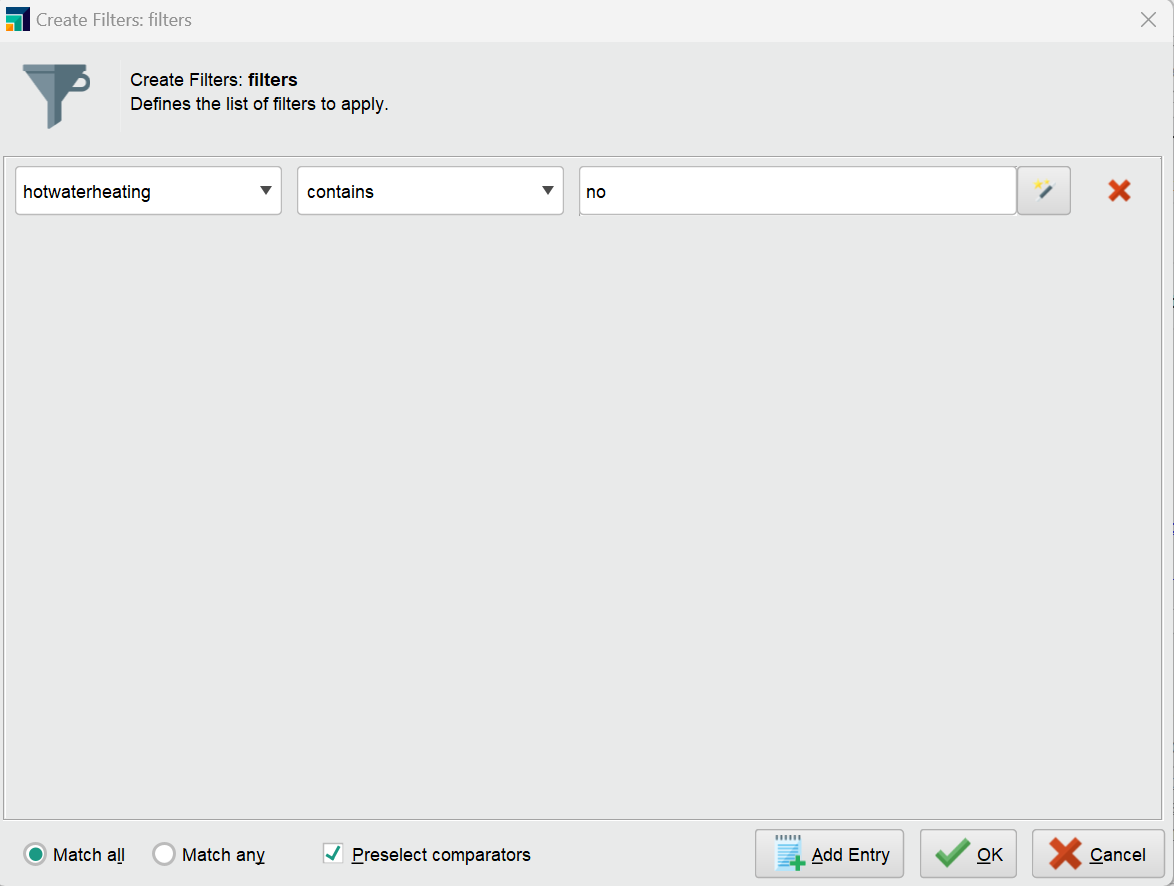
****

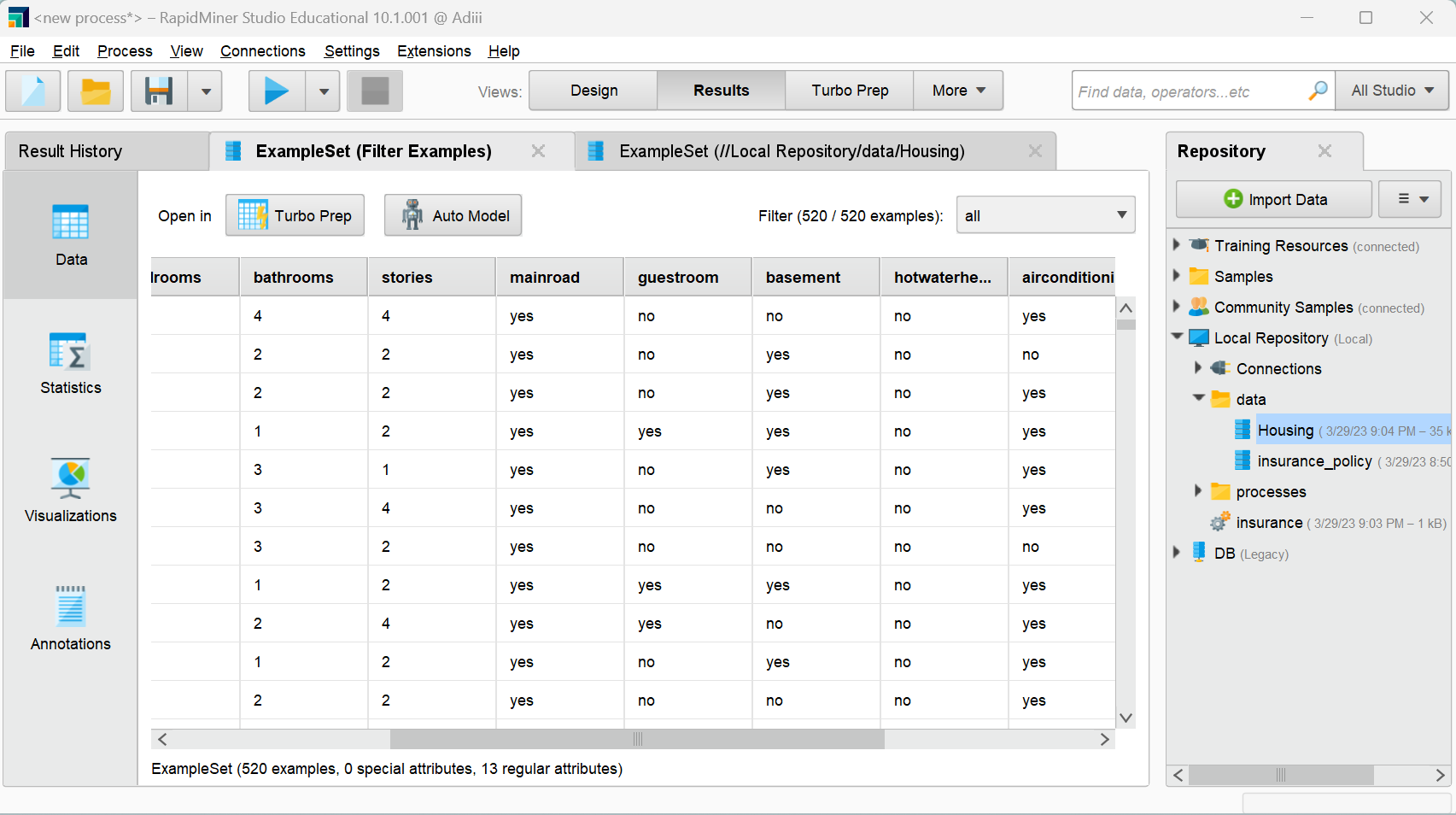
**Implementation:**

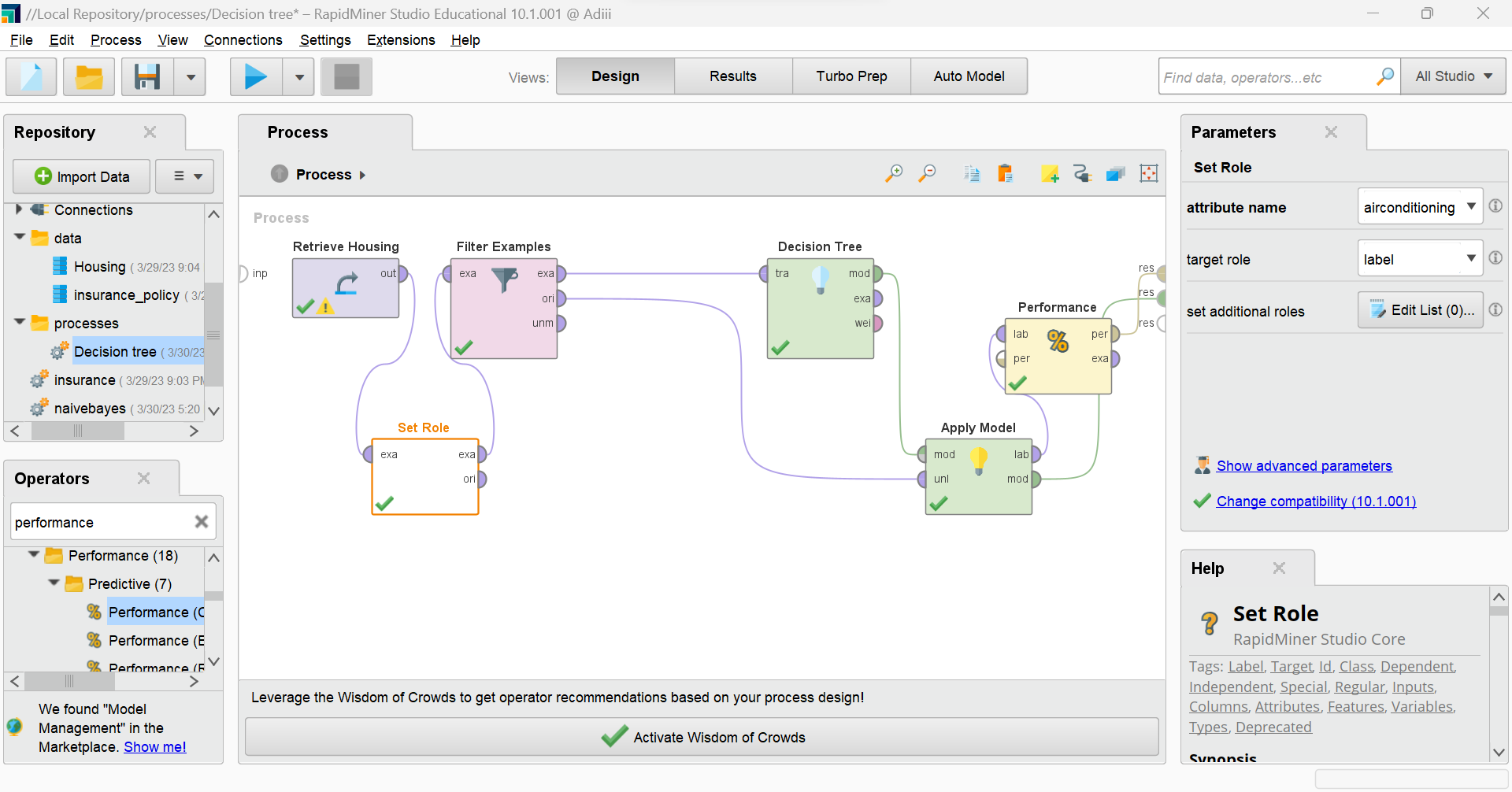
Decision tree

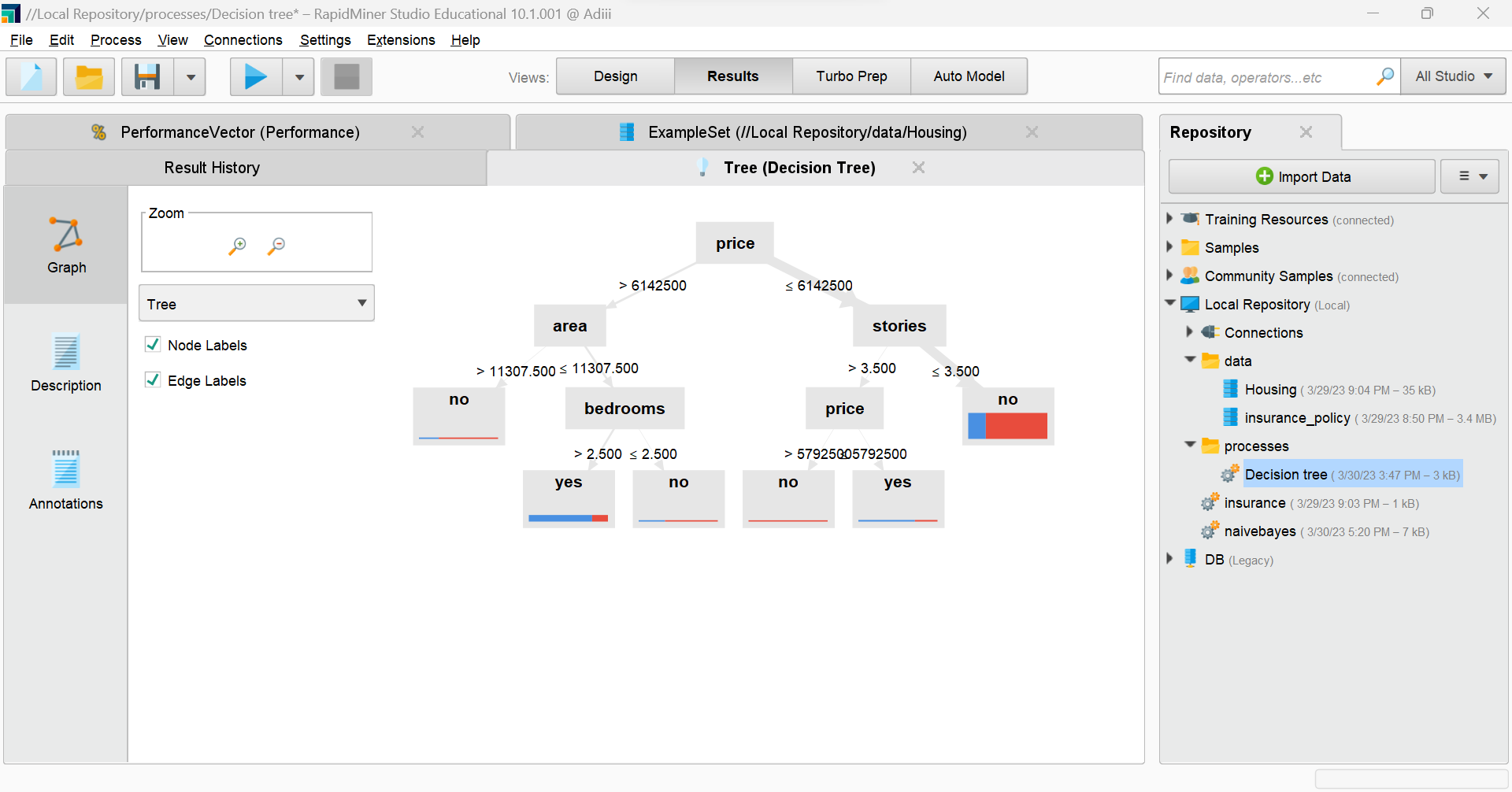




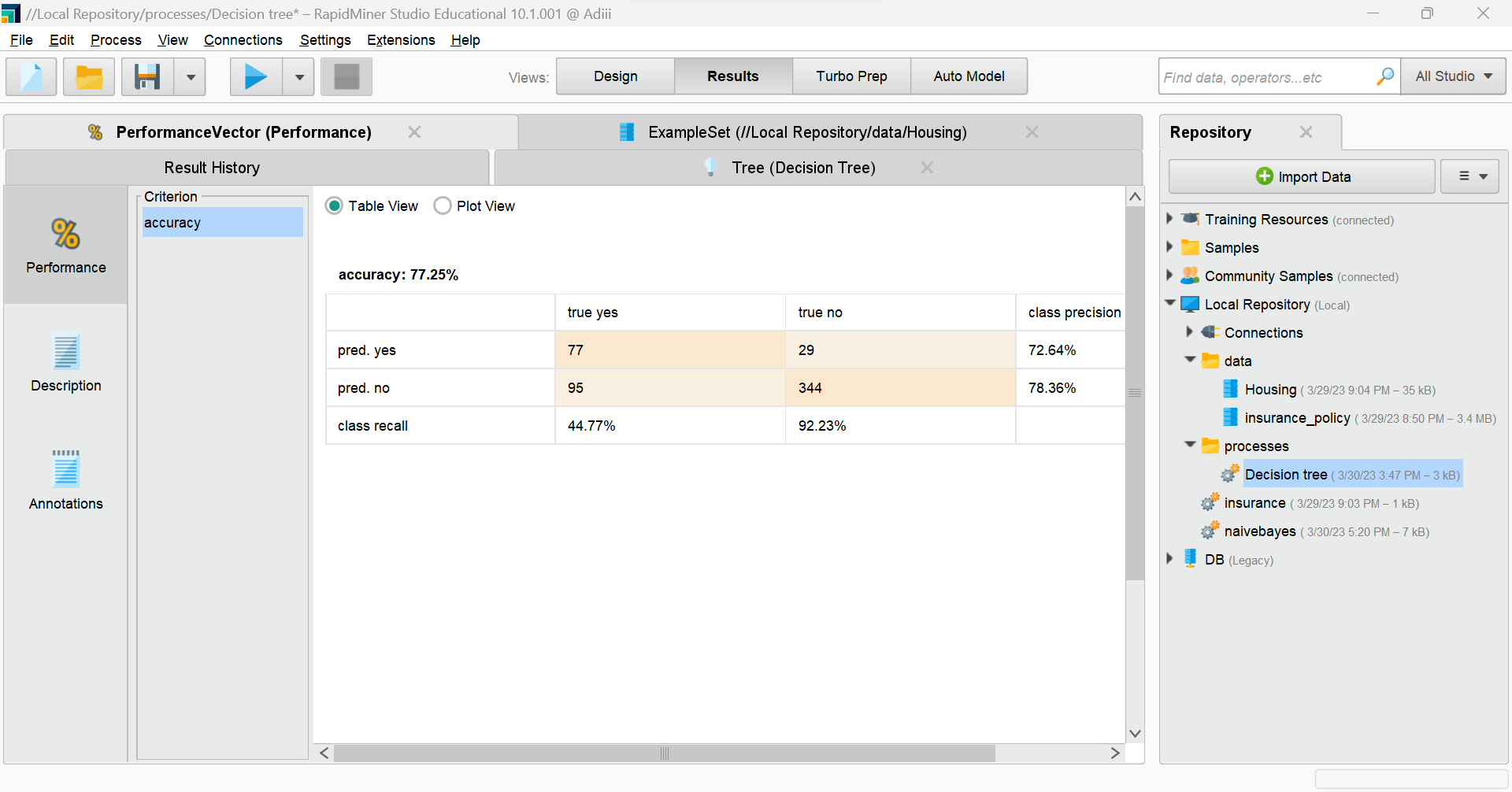


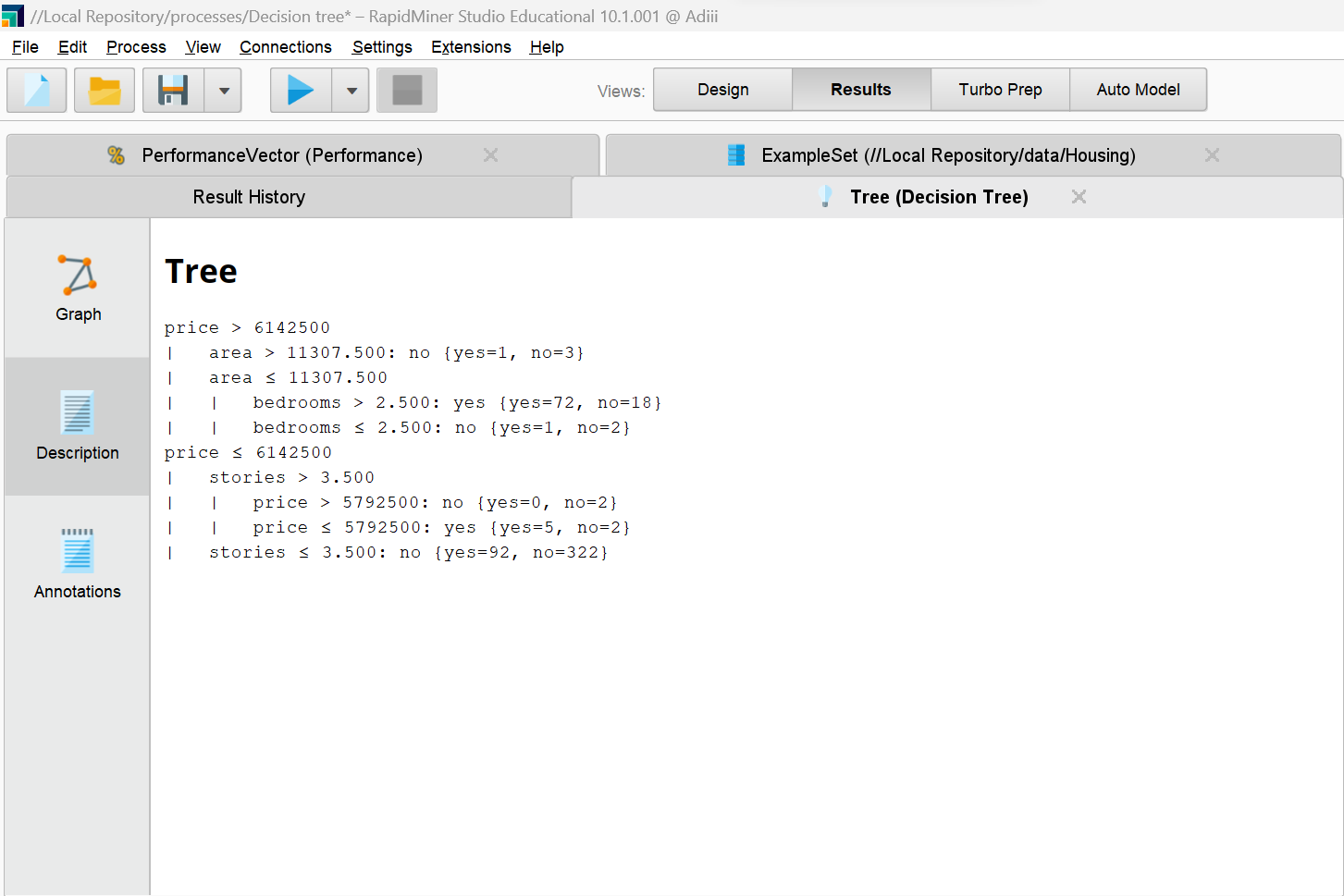




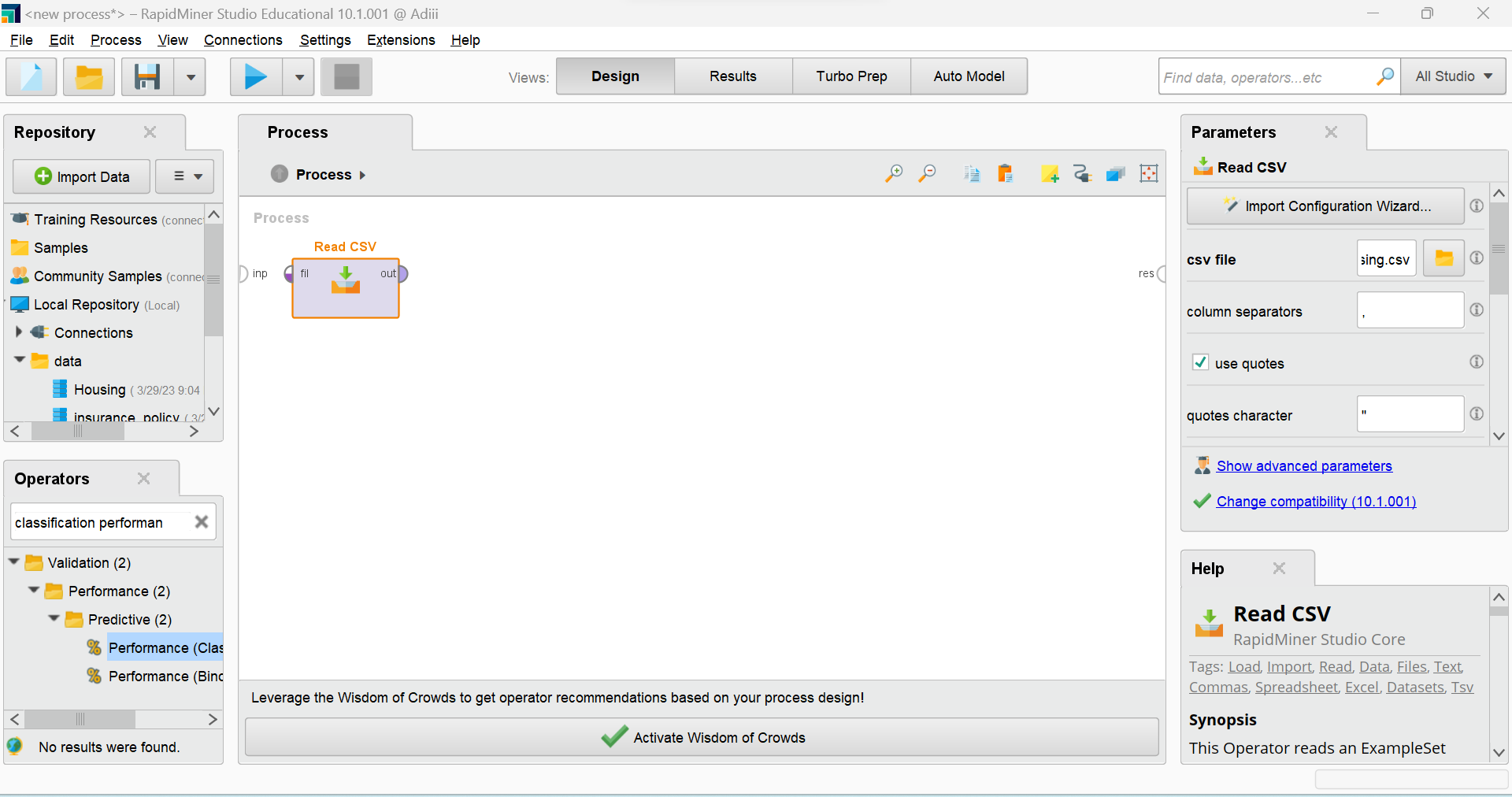


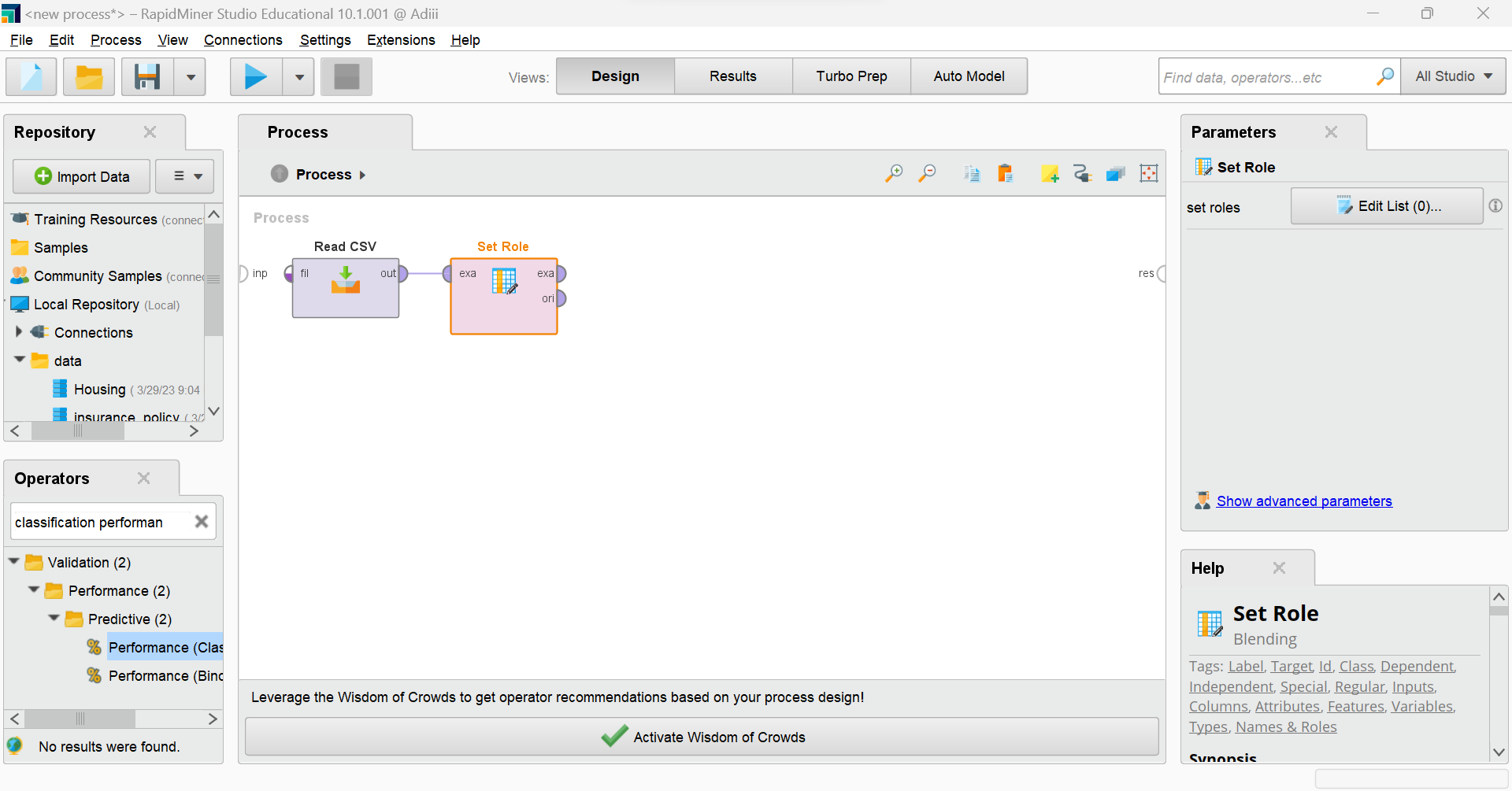


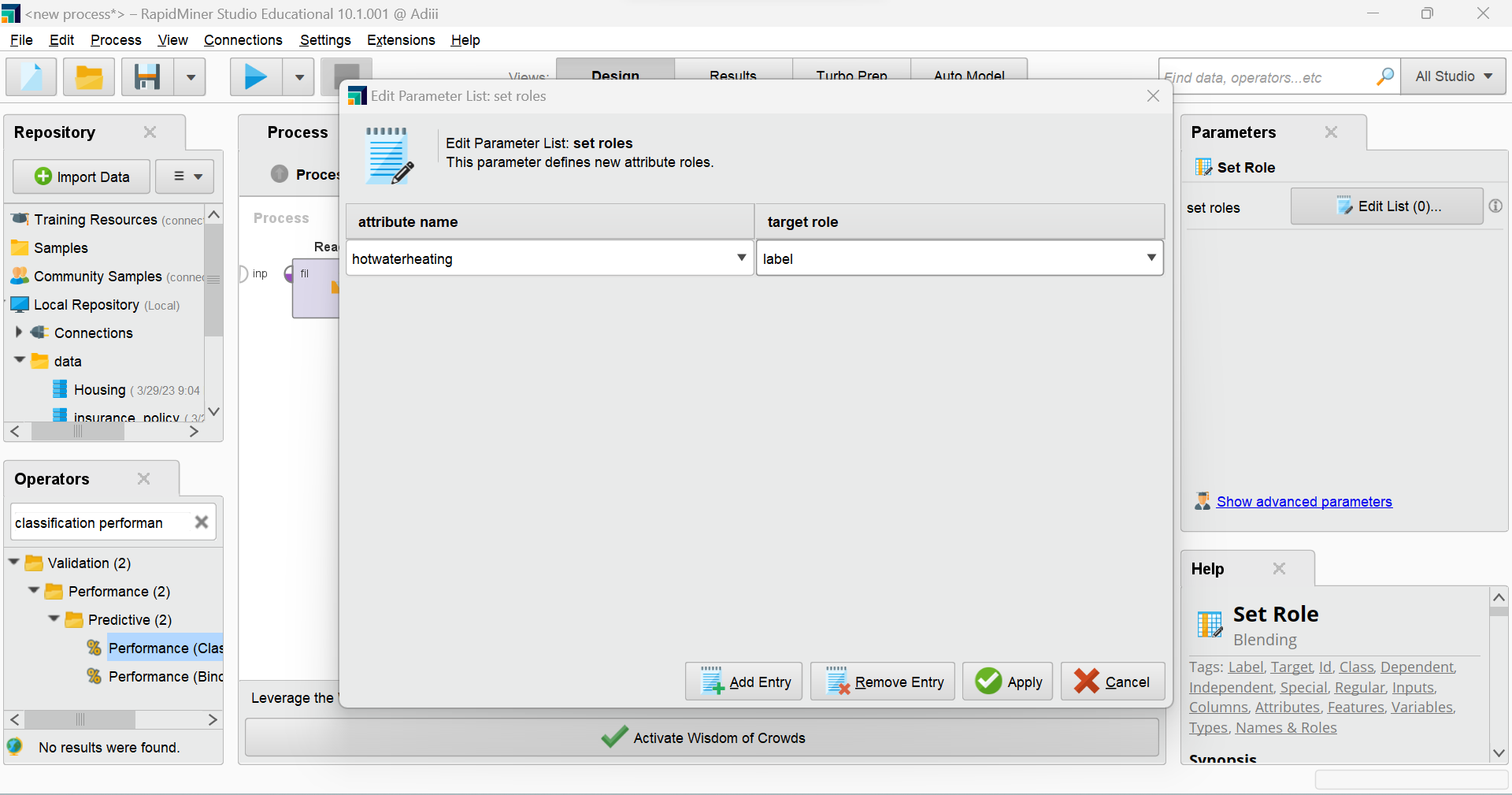


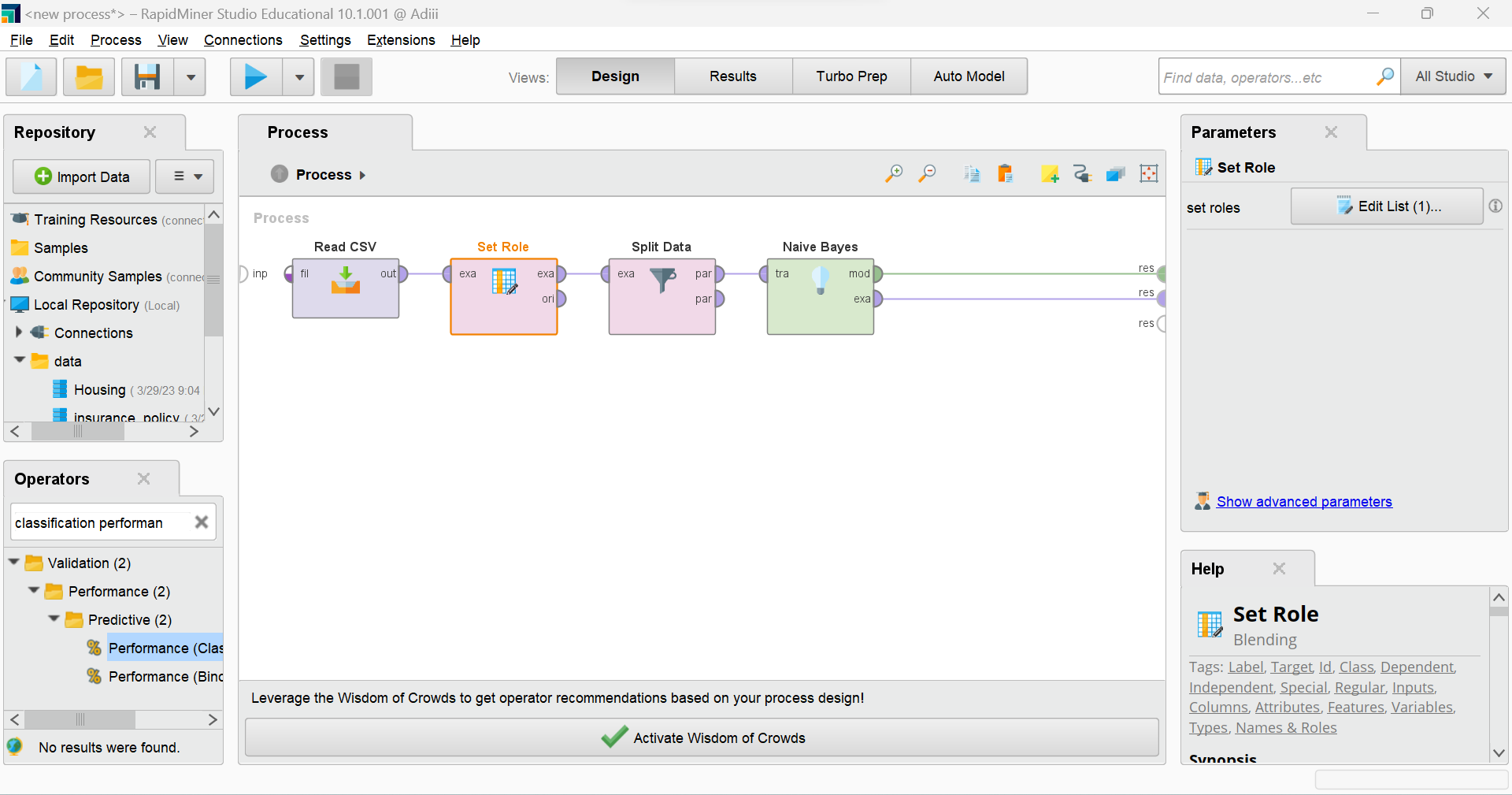


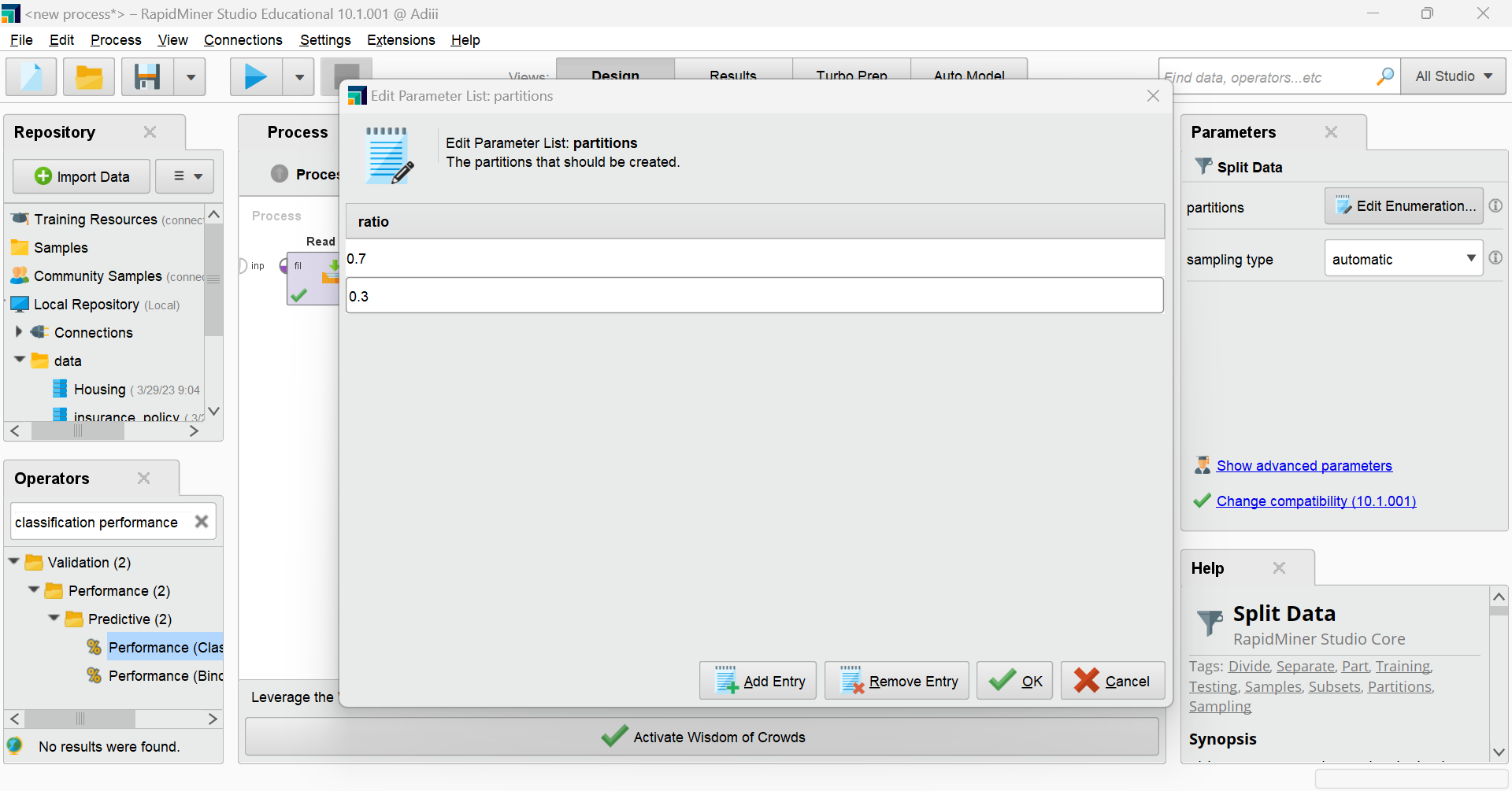
Naive Bayes

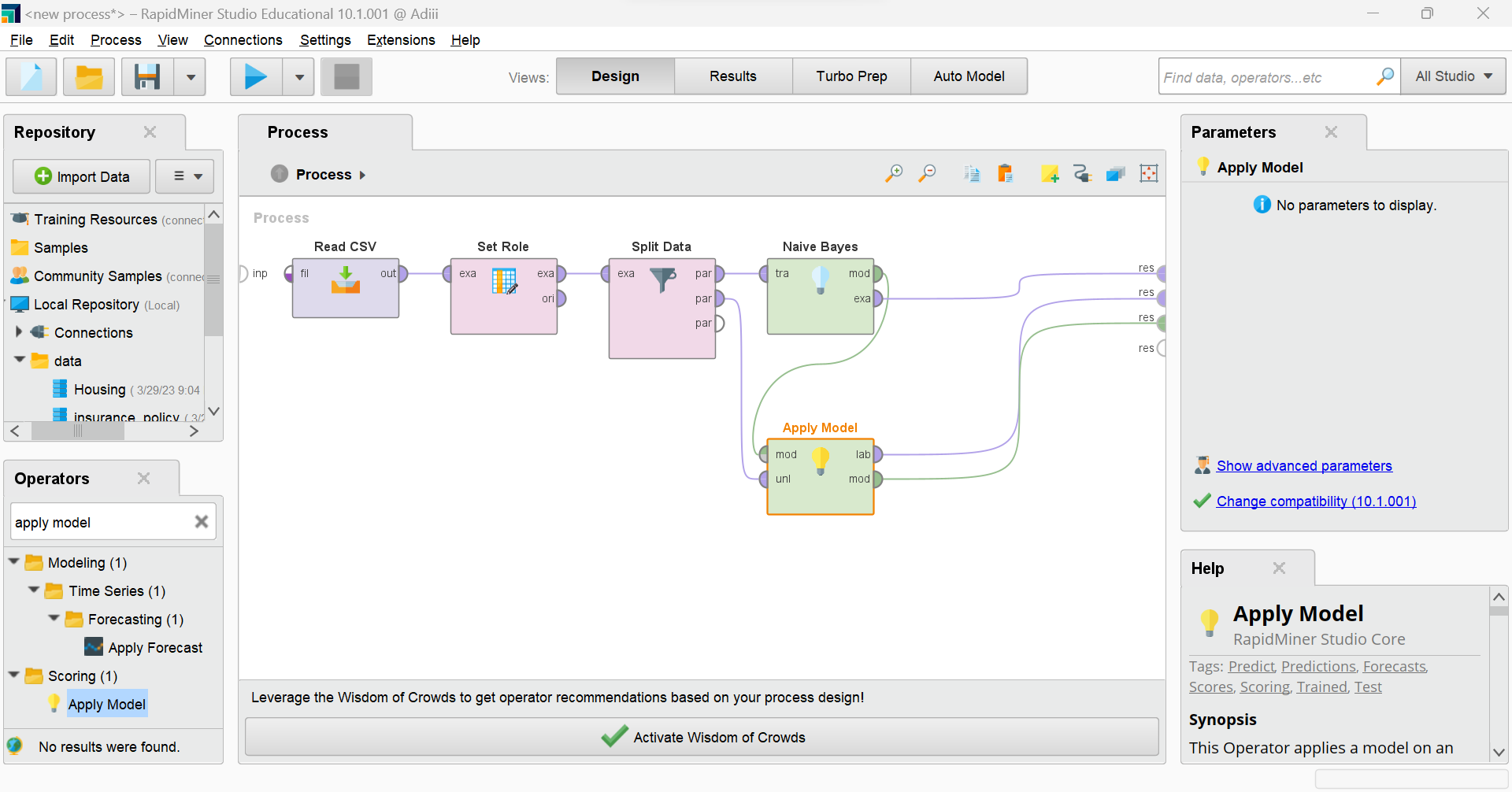


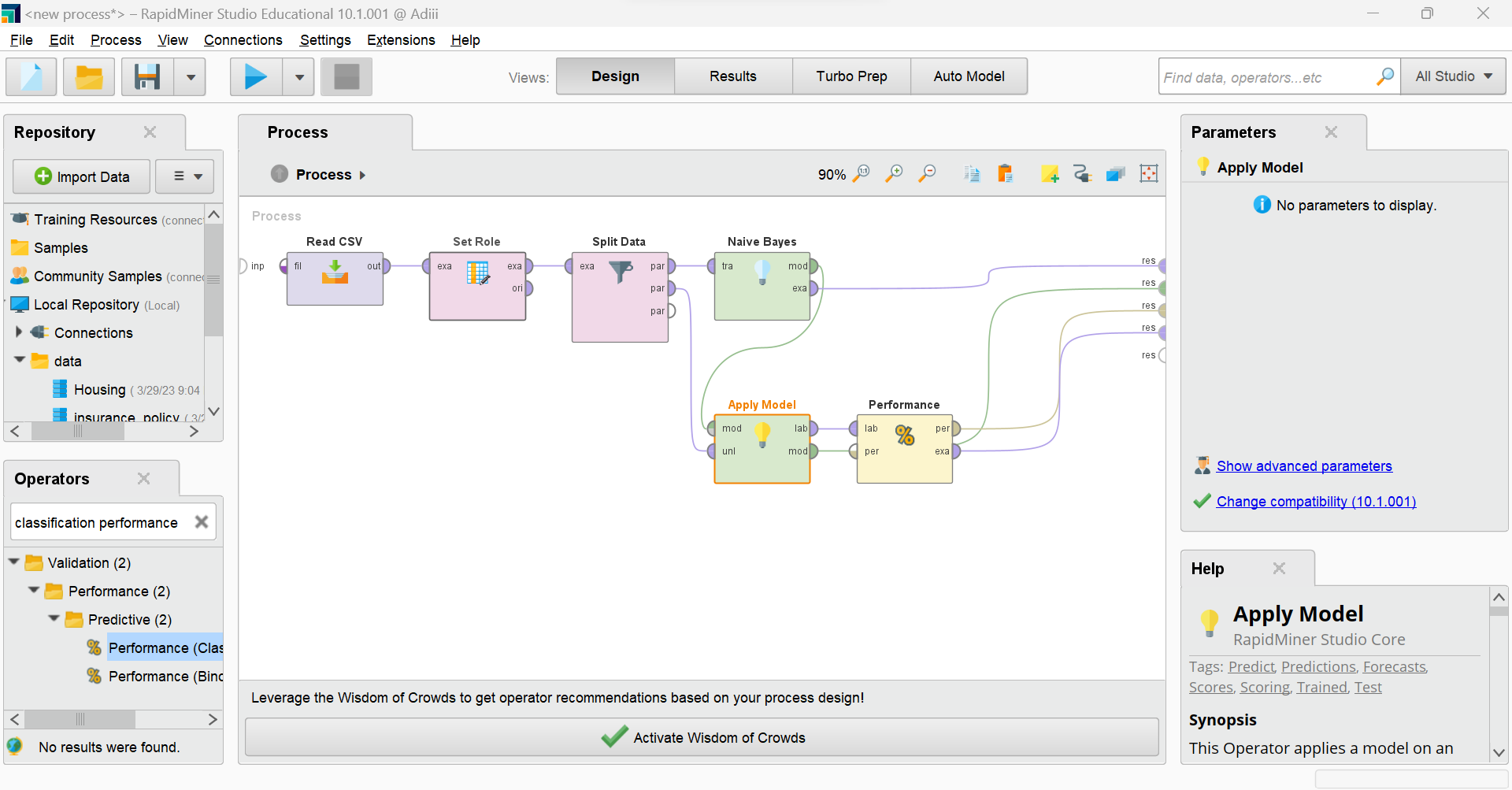


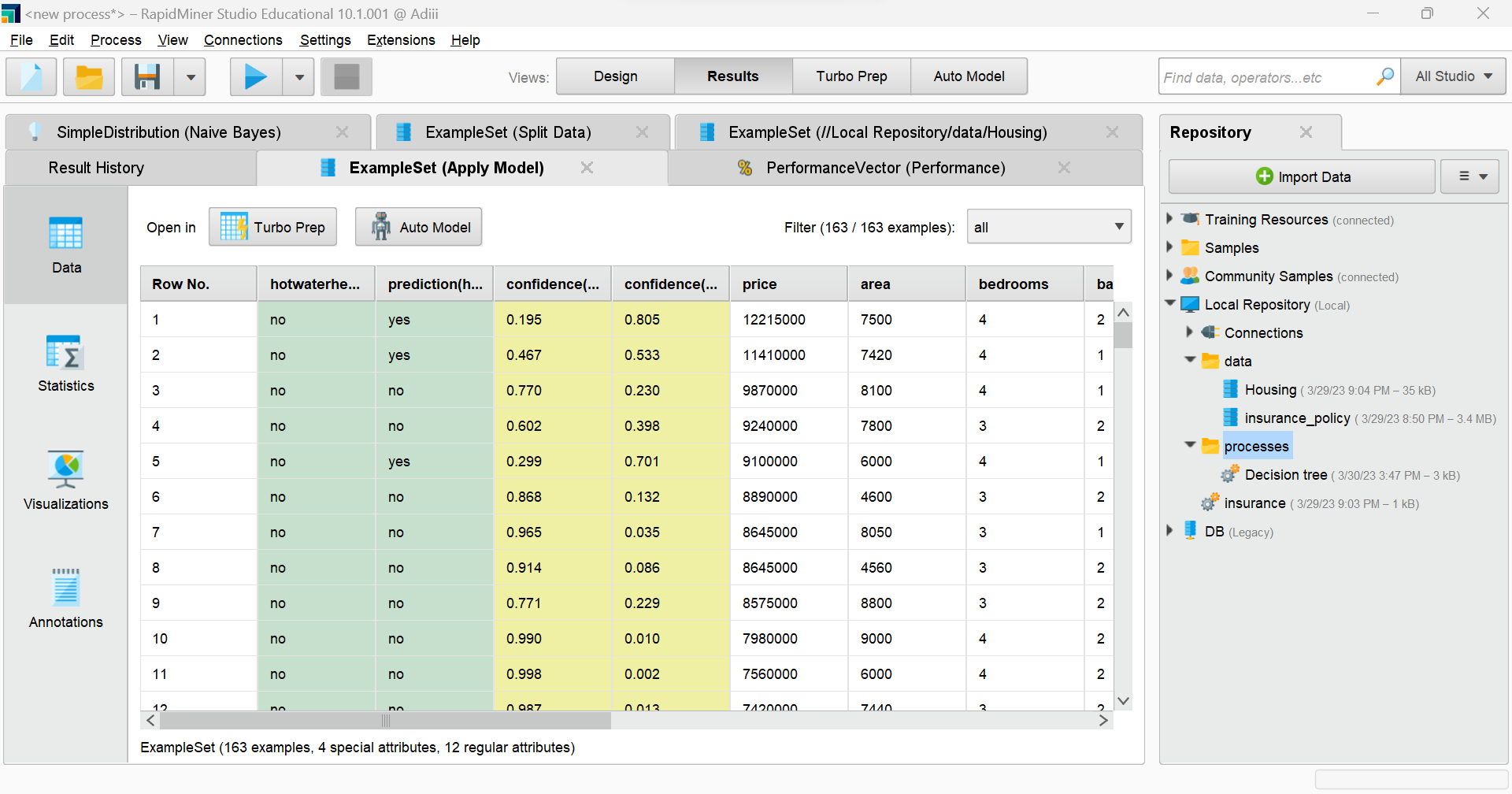


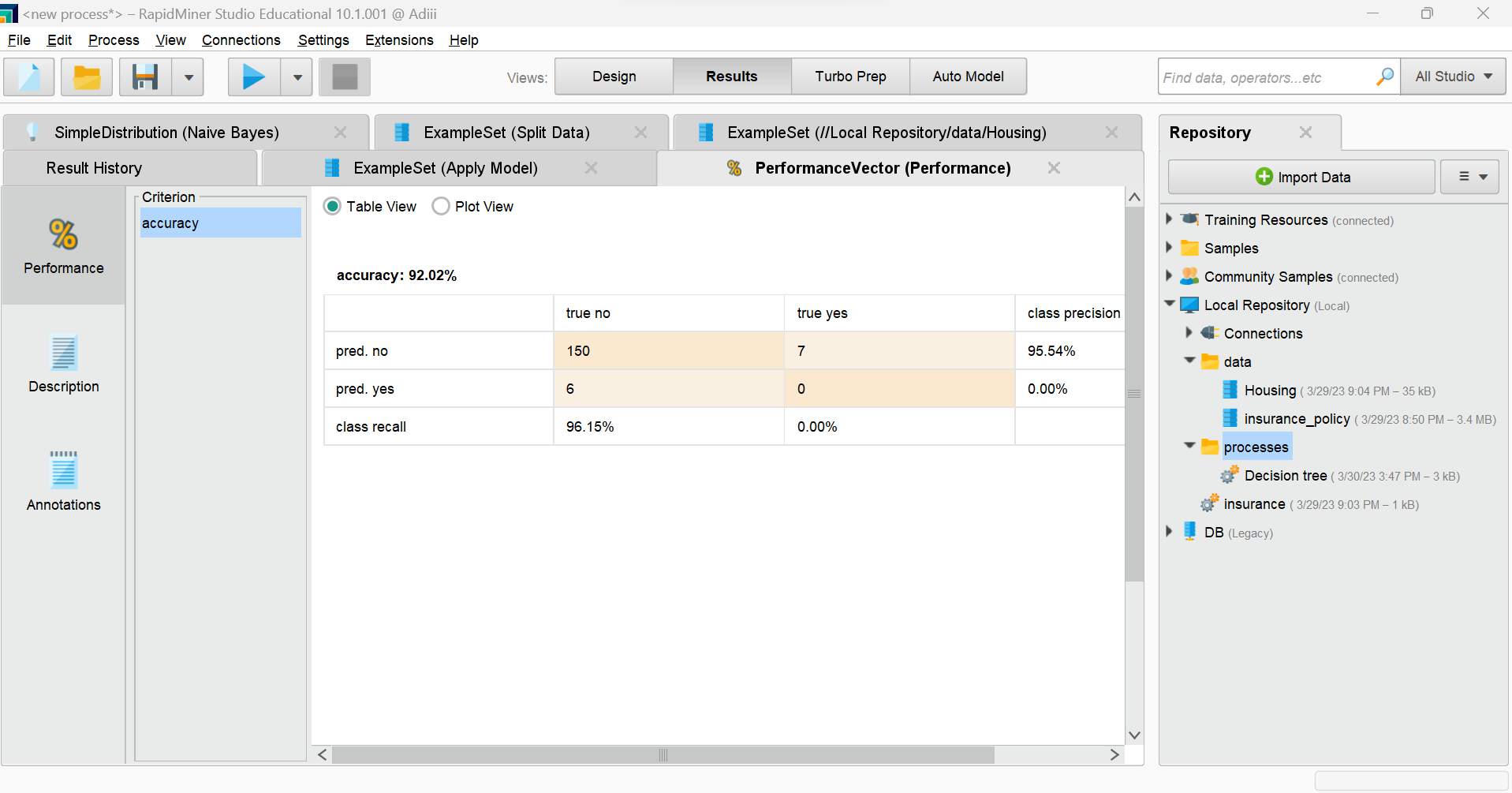


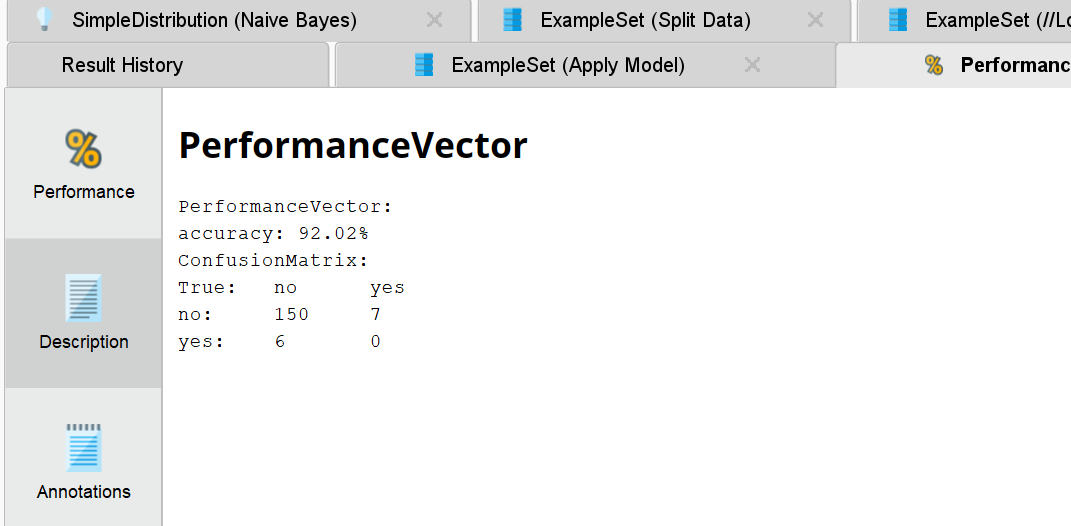


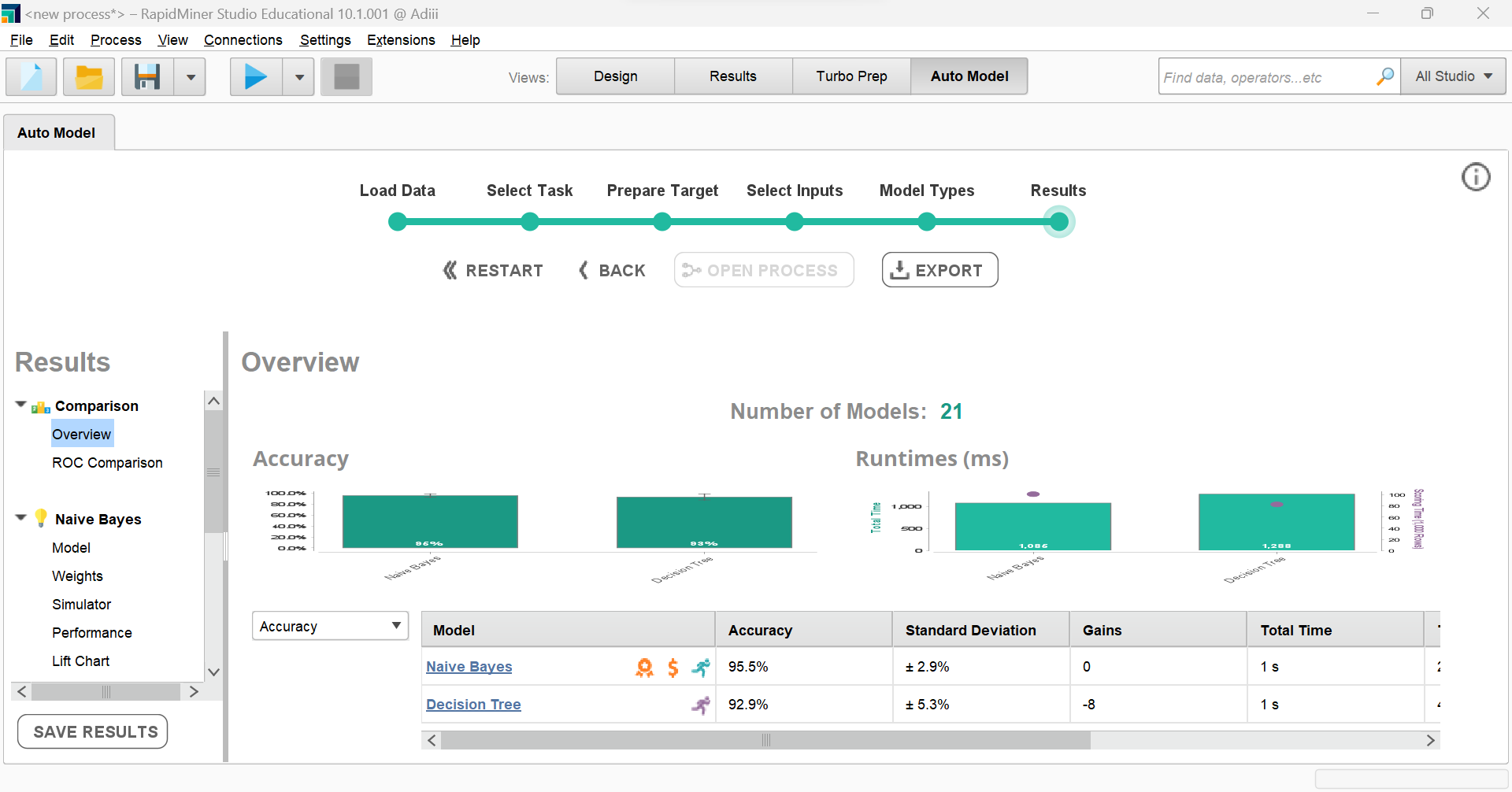


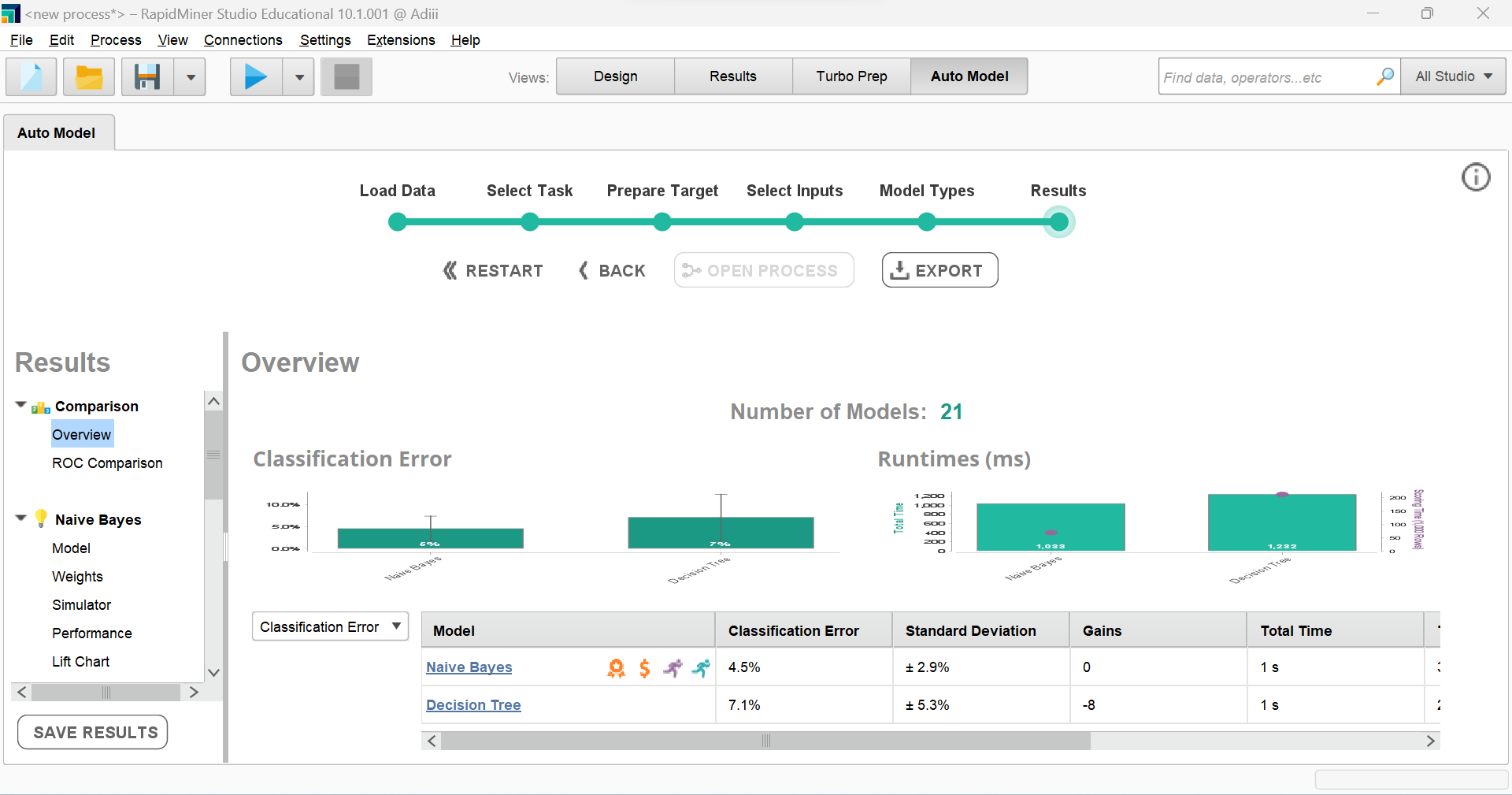




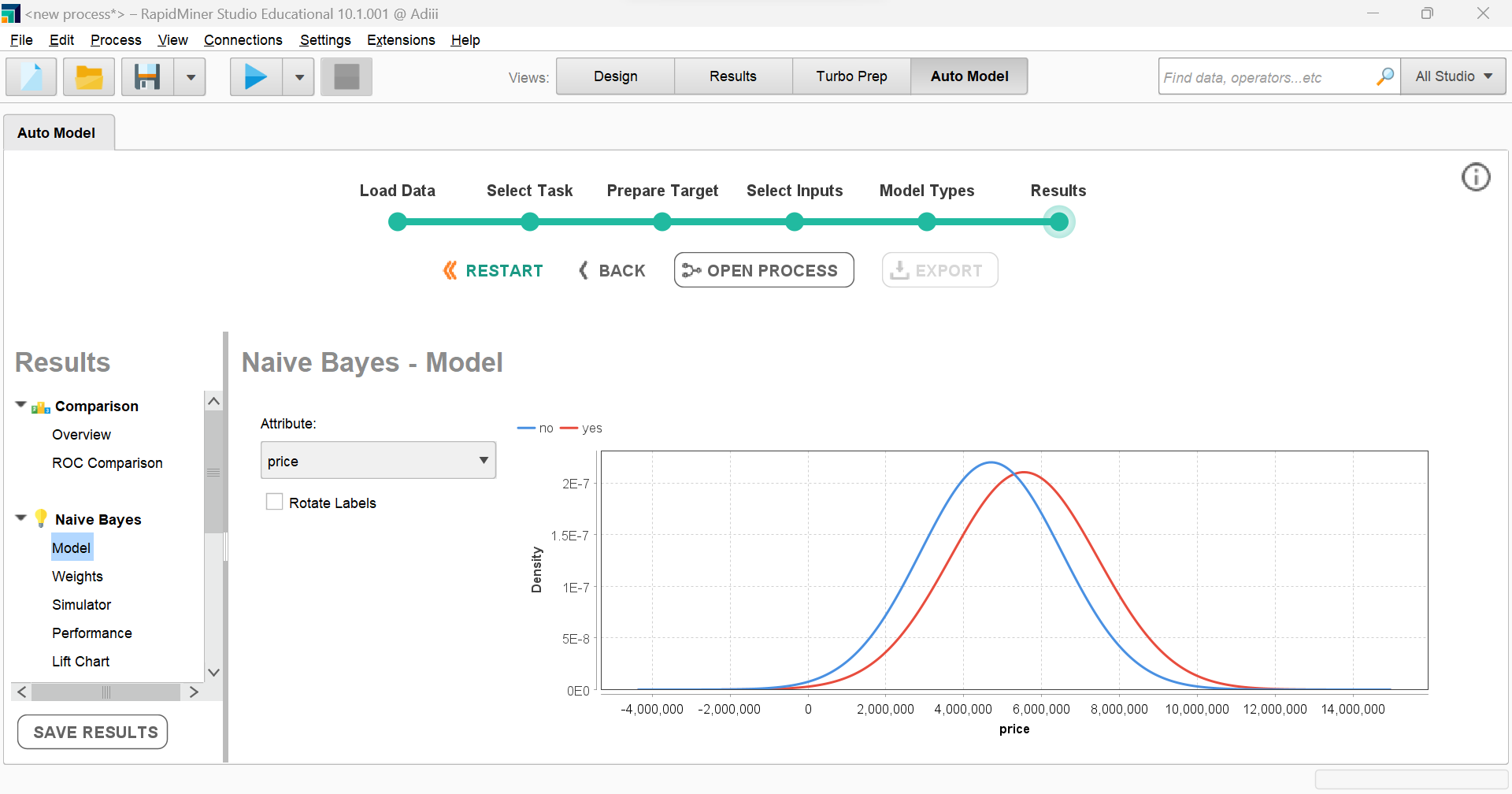




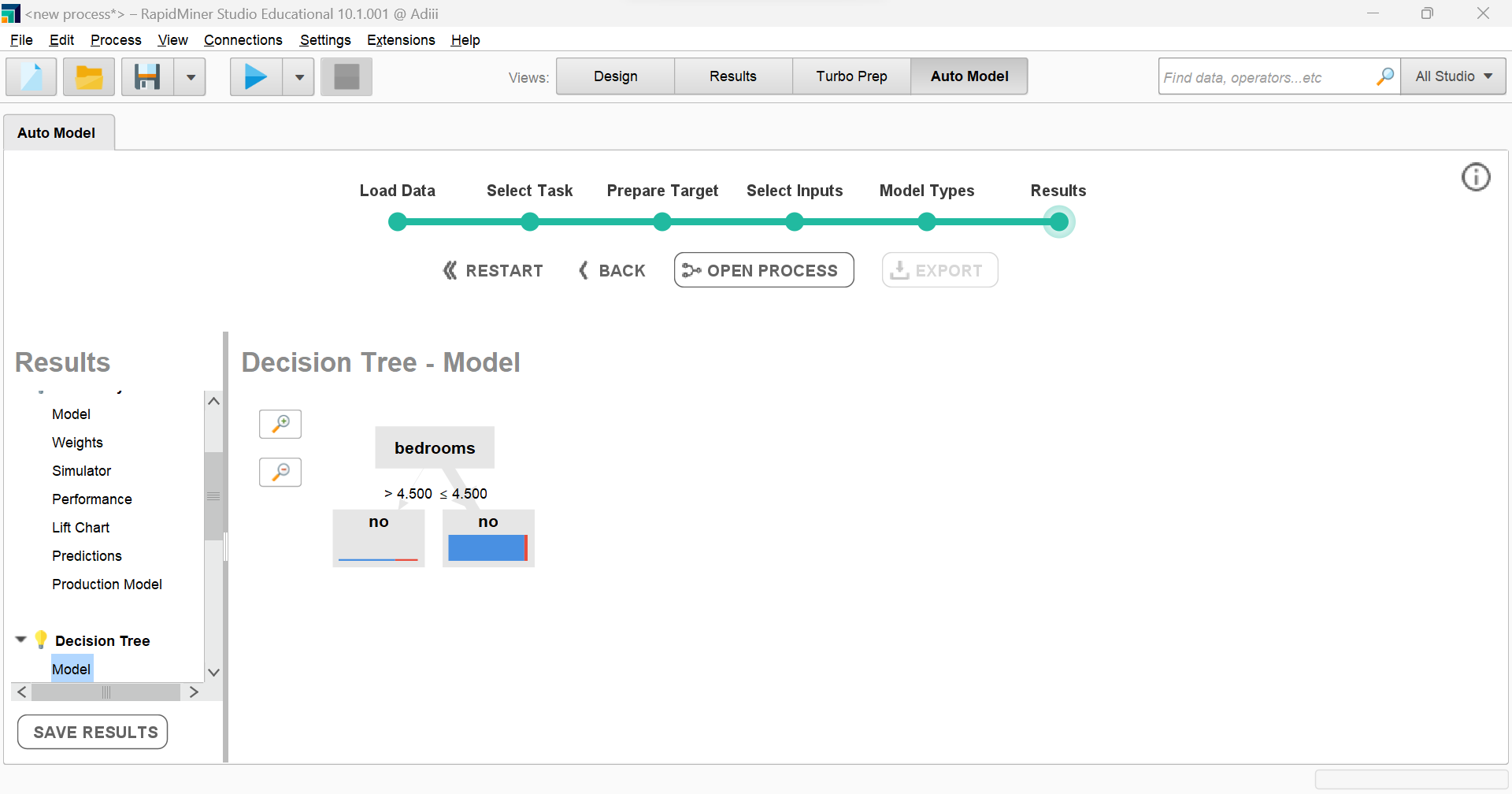




Naive bayes



Decision tree



**Conclusion**-

We have successfully explored Rapid Miner and implemented classification models like Decision Tree and Naive Bayes.