

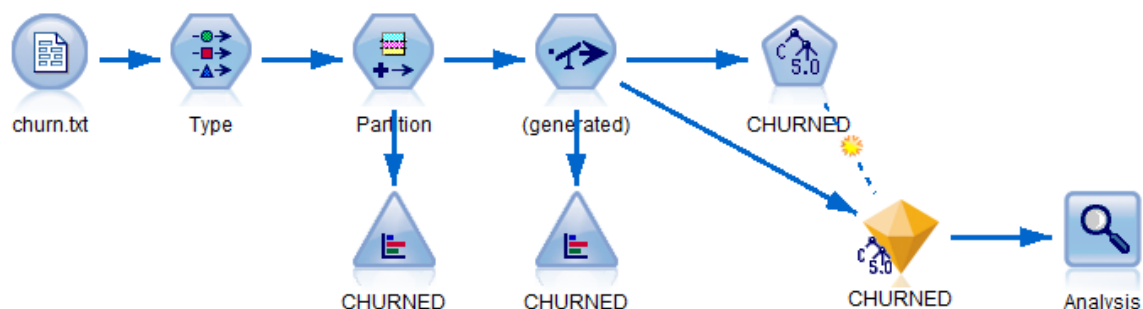
Classification with Unbalanced data

In a classification problem, where there are one, two or more classes, it is important that there are sufficiently many examples of each class, so that the learning algorithm can learn the concepts well enough. Often this is not the case, and the dataset we need to work with may have a **minority** class with much fewer examples than the other classes. In such a situation, it may be helpful to employ an

- **Undersampling technique:** Suppose there are two classes, one with 20,000 samples (minority class), and another class with 1 million samples (majority class). Building a model with all the available data may not do justice to the minority class. In this case. It is possible to take a random sample of records from the majority class (ex: 50000) and then build a model. Such an approach improves the ratio of minority class to majority class and may lead to a better model than a model built without undersampling.
- **Oversampling technique:** An alternative is to oversample the minority class. In such a technique you add synthetically created records of the minority class to increase the number of training samples of that class. There are many techniques that can be used to generate synthetic records of the minority class. SPSS modeler supports two techniques, both of which are described in this document.
 - **Add multiple copies of the minority class records:** The **Balance** node in SPSS modeler can be used for this purpose.
 - **SMOTE (Synthetic Minority Oversampling Technique):** For each training sample of the minority class, find its **k** nearest neighbors, and then randomly sample records from that neighborhood. In SPSS Modeler, a node called SMOTE implements such an approach.

There is no assurance that these techniques will improve classification accuracy or precision/recall of any class. Sometimes they do. When using oversampling techniques, it is important that the synthetically generated records are added to the **TRAINING data** only. Another way of saying this would be Do NOT use fake data when evaluating the model.

The following stream shows how to add multiple copies of the minority class records.



Churn.txt after adding a balancing node to increase the **number of training samples** of the minority class(s).

Balancing node in SPSS Modeler:

10.7 Balancing Data

In some situations you may not have a large enough data file to over-sample, but the data may still have too few records in certain categories of the outcome field. As an alternative, the Balance node can be used to make the distribution of a categorical field more equal.

Balancing is carried out by duplicating and/or discarding records based on the conditions you specify. Records for which no condition holds are always passed through. The duplication of records is literally that. If a category (condition) has 100 records and it is duplicated by a factor of 3, there will then be 300 records in that category. Discarding records works in reverse by literally dropping some of the records which meet a specified condition.

There is no free lunch, though. You can increase the number of records in a category, such as customers with 0 loans, but you are not increasing the amount of information therein, simply adding copies of existing records. But when a dataset is very small, or the number of records in specific categories is too few, balancing may be the only method that will allow you to develop a reasonable model. As with over-sampling, you must validate any model developed with balanced data on a data file that matches the population distribution of the outcome field.

The Balance node is located in the Record Ops palette, but we won't need to create one from scratch. A Balance node can be generated directly from a Distribution table (or even a histogram, as continuous fields can also be balanced, although this is less common).

We'll balance the credit data on *loans*.

Run the Distribution node named *loans* attached to the Source node
In the Distribution chart window, click **Generate...Balance Node (boost)** (not shown)

A node named *(generated)* has been added to the upper left of the Stream Canvas

Attach the *(generated)* node to the Source node
Edit the *(generated)* node

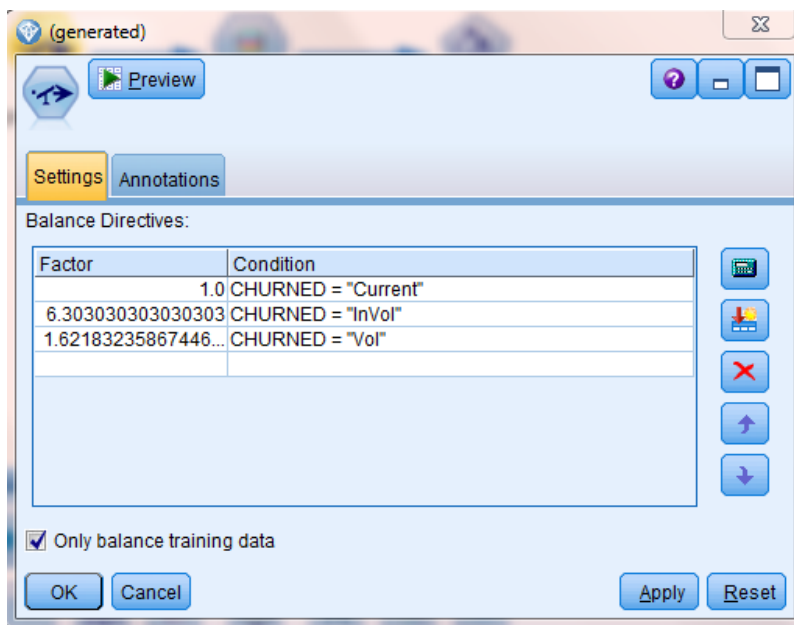
In the familiar example of Churn.txt, after adding the type node and specifying type, add a distribution mode (from Graphs palette), select CHURNED as the field to check distribution and run it. You will see the distribution is as follows.

Value	Proportion	%	Count
Current		56.33	832
InVol		8.94	132
Vol		34.73	513

In this example, the “Invol” class has much fewer samples than the other two classes. Next click the Generate tab (the third, after File and Edit). Open the Generate node and select the Balance Node (Boost) option. This will create a generated node and place it in the right top of your canvas.

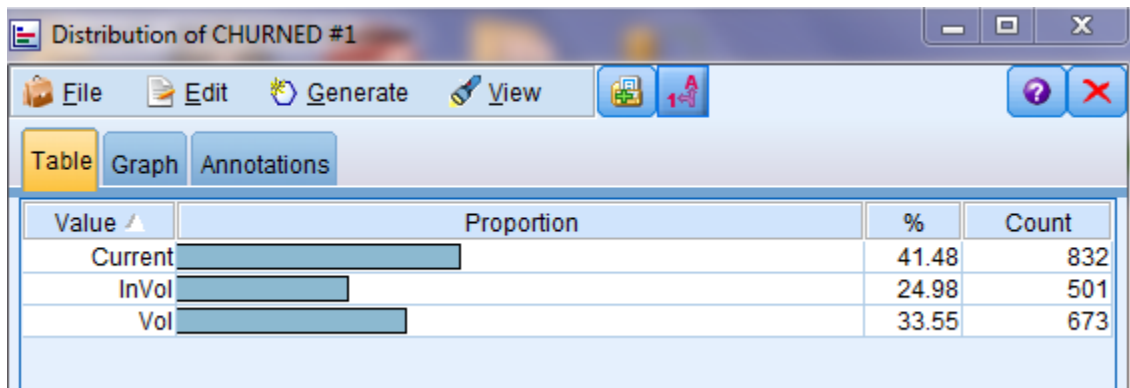


Connect the generated node to type node as shown in the stream on the first page.



Make sure to select **“Only balance training data”** option at the bottom left of the screen above. It is OK to generate additional data to train the model, but **the data for testing should not be faked!** If your data mining task is such that there is no partition node, then SPSS will ignore the option. Make sure the partition node appears before you do the balancing in the stream. Otherwise SPSS will not which records are training and which are testing.

After you run the generated node, attach another distribution node to see the current distribution.



Value	Proportion	%	Count
Current		41.48	832
InVol		24.98	501
Vol		33.55	673

Note that the number of records for 'Invol' and 'Vol' have been boosted.

The rest of the stream building and classification, and evaluation are as usual. In this example, the **analysis** node shows the confusion matrix to be as follows. Compare the performance (over all accuracy, precision and recall for all classes on test data).

There is **no guarantee** that this balancing technique will improve classification accuracy. Critics say (even the IBM manual says so) that **you may be adding more records, but there is no new information**. There are cases where the **oversampling** technique helps and other times it does not.

The undersampling technique is viable only when you have a large dataset.

Ex:

Class 1: 5000 examples

Class 2: 65000 examples

then even after undersampling class 2, you can still have 5000 examples of each class to build a model.

SMOTE (Synthetic Minority Class Oversampling Technique)

In SPSS Modeler, SMOTE is available as a Python addin.

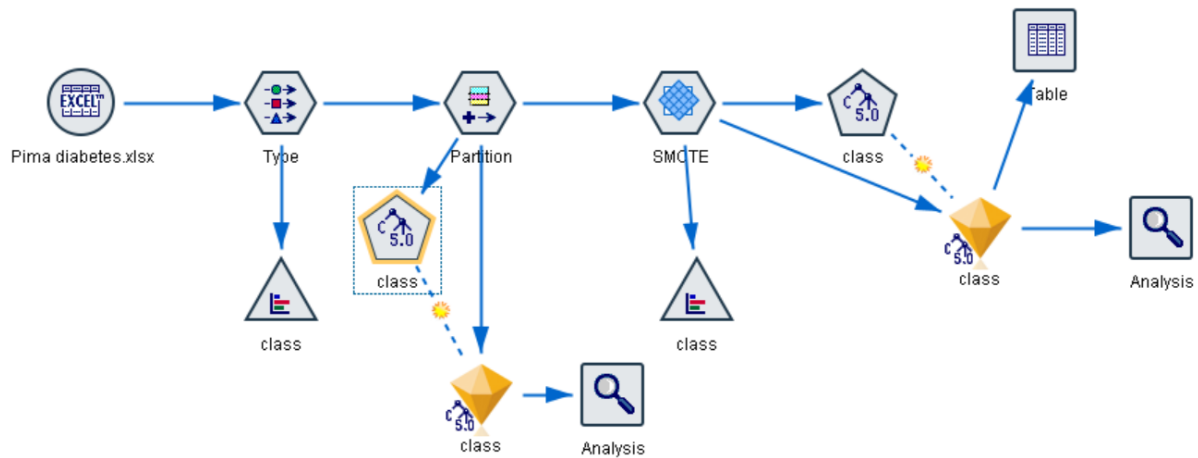


Unfortunately, SPSS Modeler documentation does not seem to have a section on SMOTE. The usage is straight forward. The original paper on SMOTE is:

N. V. Chawla, K. W. Bowyer, L. O. Hall, W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique", Journal of Artificial Intelligence Research, Vol 16, 2002.

An example stream is shown below. Again, be careful NOT to add the synthetic data for Testing. The SMOTE node comes with an option to specify this.

The following example is built on the Pima diabetes.xlsx dataset. This dataset has several features, but all of them are **continuous variables**.



Other references:

http://rikunert.com/SMOTE_explained