

EXERCISE 2

Purpose: To learn how to **oversample the number of successes in a classification problem with very few “successes”** using XLMiner ©.

Go to the website for this course and download the file “catalog_multi.xls.” This file should also be one of the files in your example dataset subdirectory in XLMiner. Use this file to complete the following tasks. Hand in your work on Wednesday, February 3 in class. In this exercise use the random number seed 12345 so that everyone will get the same answer.

- a) Put all of the listed variables in the “Variables in the partitioned data” window. This will allow you to begin the oversampling process.
- b) Choose as the target variable “buy” as the “output” variable. You will notice that this variable has two classes (1 = buyer, 0 = non-buyer).
- c) Since the number of buyers is only a small fraction of the number of cases that we have (0.9896% of 58,205 = 576 buyers) we need to build a training set that has 50% of the buyers in it, i.e. oversampling of the buyers to put into the training data set. Therefore, specify the % successes to be put into the training data set to be **50%**. Just for your information, **in the case of very few successes in a data set, Classification models are better built on oversampled training data sets. However, we will eventually want to evaluate them on data sets that reflect the actual population proportions of successes and non-successes.** (We assume that the sample proportions we drew in our original sample are representative of the actual population proportions.)
- d) Furthermore, if we want to get an **unconditional** read on how accurate our better classification methods are in correctly classifying buyers in new data sets we need to set up a test data set. Now the oversampling algorithm in XLMiner sets up the validation data set to replicate, as closely as possible, the proportions of buyers (0.9896%) and non-buyers (100% - 0.9896% = 99.0104%) of the entire sample while, at the same time, using all of the remaining successes (buyers) that were not chosen in the construction of the training dataset (i.e. $576/2 = 288$). Therefore, to be representative of the entire sample, we need to randomly assign X number of non-successes to the validation data set where $0.009896X = 288$. Obviously, $X = 29,102$ is the number desired. Now, if we want to split the validation data set into two parts ($29,102/2 = 14,551$ cases) to form a test data set roughly having the overall sample proportion of successes as well, then you need to choose **50%** for the “**take away percentage**” from the validation data. This will, of course, result in 14,551 cases with roughly 144 successes in each of the “population representative” validation and test data sets.
- e) After having made these choices, execute the oversampling in XLMiner and cut-and-paste on a separate sheet to hand in **only the “results” (top) part** of the

“Data Partition with Oversampling” sheet. (Be careful and don’t try to print out the entire sheet as you will eat up a lot of paper when the printer tries to print out all of the $576 + 14551 + 14551 = 29,678$ cases that have been chosen out of the original 58,205 cases.

- f) To convince me that you have gotten the right partition of the data, print out separately the first five observations of the training data set, the first five observations of the validation data set, and the first five observations of the test data set and hand them in. In printing these observations out you only have to print out the first, say, 5 columns of the observations.