

SESUG Paper 103-2019
Hybrid Rare Event Sampling Technique
Ross Bettinger, Consultant

Abstract

We discuss a hybrid sampling methodology for building and validating machine learning models for classification for which the target event is a small proportion of the total population. This methodology may be useful for fraud detection or similar applications for which the response variable is typically scarce. We call this variation of cross-validation methodology the HYbrid Rare Event Sampling (HYRES) technique because it is meant to detect rare events with high resolution through sampling.

Keywords

Classification, cross-validation, imbalanced classes, machine learning, model building, model validation, rare event, sampling methodology, SAS Enterprise Miner®, SMOTE, threshold-free performance metric

Introduction

We once had a fraud detection problem involving 2,000 observations labelled Fraudulent out of a total population of 1,000,000 observations, a 0.2% event rate. Our immediate thought was to stratify the sample into equal numbers of fraudulent and non-fraudulent observations, build models on the biased sample, and use prior probabilities to correct for sampling bias. But given the typical requirement of creating training/validation/test¹ datasets (say 70%/20%/10% proportions), will only 1,400 training events represent sufficient and consistent behavior containing enough features to represent fraudulent behavior when the target population is so small? There is a pronounced imbalance in the class frequencies of the events and nonevents, so we must approach the disparities carefully.

If we use bagging, we have to select fractions of the total population (or even of a sampled population) and build unbiased or biased models. If we build cross-validation models, we have the same problem. When assembling the training data, we typically select subsets of the fraudulent observations and combine them with subsets of the non-fraudulent observations. Regardless of the algorithm, we do not use *all* of the fraudulent observations in each sample of the training data because of the model building methodology. There must be a set of validation data against which to test the model's performance during the training process. Thus, we may weaken the model's power of detection at the outset because we withhold a subset of the event data for evaluation of model performance. If we create a test/holdout dataset for evaluation, we further deplete the number of observations available for model-building.

Our primary problem is: how do we use as many of the infrequent examples as possible in the training process without resorting to oversampling, SMOTE-based over- and underrepresentation, or inverse probability weighting?

Figure 1 illustrates the cross-validation strategy [1, 2]. Each sample uses a fraction of the potential maximum number of observations proportional to the T/V/T ratios. The training, validation, and test (holdout) datasets containing 2,000 rare events are labelled S_T , S_V , and S_H and the nonevents are not shown because their frequencies may vary, depending on the sampling proportions of event/nonevent.

¹ To be concise, we will refer to "training/validation/test" as "T/V/T" in the sequel.

3-fold Cross validation

10% Test	20% Validate	70% Train
67 S_H	133 S_V	467 S_T
66 S_H	134 S_V	466 S_T
67 S_H	133 S_V	467 S_T

Figure 1 Standard 3-fold Cross Validation

Aside from the test sample which is not used in model construction², if we use fewer than all of the remaining [fraudulent] cases (e.g., 70% for the training dataset, 20% for the validation dataset) in a model, we may be losing important information since we are trying to identify infrequent patterns, and each scarce event is valuable because it represents an example of the behavior that we are trying to detect.

So, we devised the following sampling construction methodology:

- Let us label the set of fraudulent observations as Signal (S) and non-fraudulent observations as Noise (N).³
Then we want to assign labels “S” to observations S in {S, N} in such a way as to optimize some appropriate performance metric such as area under the ROC curve (AUROC) or lift or other KPI.
- Let S_T be the subset of S observations to be used in the training sample, let S_V represent the subset of cases available for validation, and let S_H represent the $\{S - S_T - S_V\}$ subset of cases available for testing model performance (holdout sample).
- Similarly, let N_T , N_V , and N_H represent subsets of Noise observations.

² Training data are used to compute a model’s parameters, and validation data are used to produce error estimates which provide feedback in computing model parameters. Since the model uses the validation data in the learning process, we cannot consider the validation data *per se* to be independent of the model building process. The test dataset represents the data that will be scored by the model once it is deployed for use. It is independent of the model construction effort and is used to evaluate a model’s performance after the model has been built on the training and validation datasets. Hence, we must use the test data as a holdout sample to evaluate model performance without bias.

³ We are using the language of statistical signal detection theory when we assign the labels “S” and “N” to event observations and nonevent observations because we are trying to recognize patterns in the data that allow us to discriminate between a signal (Fraud) and the background noise (Nonfraudulent) in which the signal is embedded.

- We compute subsets of N_T as N_{T1}, N_{T2}, \dots to contain disjoint samples from N_T . We also sample to compute subsets for N_V . We do not use N_H in the model-building process, only in evaluation.
- We build models from training datasets $(S_T, S_V, N_{T1}), (S_T, S_V, N_{T2}), \dots, (S_T, S_V, N_{Tn})$ with corresponding noise datasets for training and validation N_{T1}, N_{V1} , et cetera. We use *all* of the S_T, S_V subsets of observations per model and the several disjoint noise subsets N_{Ti}, N_{Vi} per model to effectively hold the Signal constant and vary the Noise during the model-building process.
- We use an appropriate ensemble technique to combine the results of the models' predictions.
- We score the ensemble model using the S_H and N_H data, and we evaluate KPIs appropriate to our problem's knowledge domain.

Figure 2 illustrates this strategy. Each dataset uses the maximum number of rare events. The noise subsets N_{Ti} and N_{Vi} , which are not shown in the figure, are varied for each model.

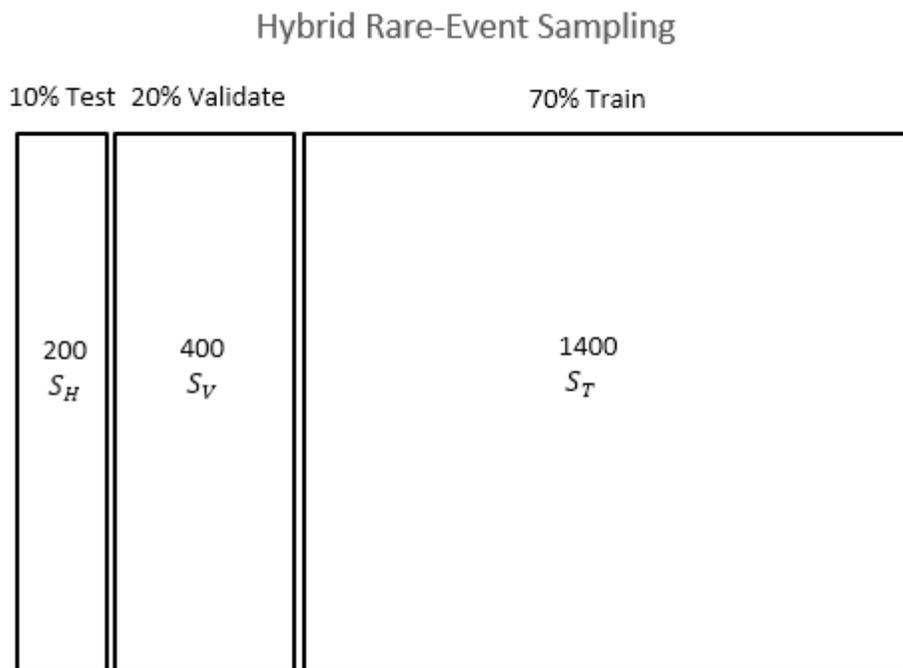


Figure 2 Hybrid Rare-Event Sampling

While we may, in theory, assume that all of the data are homogeneously distributed over all of the samples selected, it is good practice to vary the random number seed(s) used in selecting the training, validation, and test datasets and to build several models with the resampled data to ensure that our results are replicable and not merely due to serendipitous happenstance.

This technique is different from n-fold cross-validation or bagging in that it uses *all* of the target events in *every* model instead of disjoint subsets of the target events. The signal S is maintained fixed over all models built and the background noise N is changed. Essentially, we propose that the HYRES methodology maximizes the signal/noise ratio by holding the Signal information constant and varying the Noise information.

Methodology

We applied the SAS® SEMMA (Sample, Explore, Modify, Model, Assess) methodology to evaluating the HYRES approach to rare event detection.

Modeling Data

We selected nine classification datasets from the UCI Machine Learning Repository [1, 2, 3, 4, 6, 9], Kaggle⁴ [5, 8], and from Luis Torgo's "Data Mining with R: Learning with Case Studies" website⁵ [7]. We varied the number of events as a percent of the total number of observations to determine if there might be an effect due to population size, assuming that the number of patterns associated with an event would be greater in a larger population compared to a smaller one. Hence, if there are more patterns associated with an event, a learning algorithm would have more opportunity to discriminate between event and nonevent and thus distinguish between signal and noise.

Table 1 lists the significant characteristics of each dataset. Citations are indicated by numbers in brackets.

Table 1 Significant Characteristics of Datasets

Dataset	# of Obs	# of Vars	Event Class Label	Subject Area
Adult Income [1]	37,340	12	Binary indicator of income level	US Census
Bank Direct Marketing [2]	41,188	20	Binary indicator of successful acquisition	Marketing
Coverture [3]	581,012	55	Integer category of forest cover type	Forestry
Credit Card Default [4]	30,000	24	Binary indicator of default	Finance
Credit Card Fraud [5]	284,807	31	Binary indicator of fraud	Finance
Higgs Boson [6]	5,171,460	29	Binary indicator for signal or background	Physics
PaySim Fraud Detection[7]	6,362,620	7	Binary indicator of fraud	Finance
Sales Fraud Detection [8]	387,010	3	Binary indicator of fraud	Finance
Wilt [9]	4,839	6	Binary indicator for diseased tree	Forestry

Table 2 describes the event class frequencies and target values in more detail.

Table 2 Details of Event Class Labels

Dataset	# of Events	# of Events as % of Obs	Event Value
Adult Income	185	0.50%	1 if income > \$50K
Bank Direct Marketing	4,640	11.27%	1 if successful acquisition of account
Coverture	20,510	3.53%	1 if cover type = 7, Krummholz
Credit Card Default	6,636	22.12%	1 if cardmember defaulted
Credit Card Fraud	492	0.17%	1 if fraudulent transaction
Higgs Boson	583	0.01%	1 if signal detected
PaySim Fraud Detection	8,213	0.13%	1 if simulated fraudulent transaction
Sales Fraud Detection	1,199	0.31%	1 if inspection report = 'fraud'
Wilt	261	5.39%	1 if diseased tree = 'w'

⁴ www.kaggle.com/datasets

⁵ <http://www.dcc.fc.up.pt/~ltorgo/DataMiningWithR/>

Model Construction

We compared the performances of HYRES models to 3-fold cross-validation models by using SAS Enterprise Miner to produce an ensemble model for HYRES data samples and an ensemble model for 3-fold cross-validation data samples, and then we compared the performance of the ensemble models to each other.

We used the %HYRES macro (described in the Appendix) to produce, for each set of scenario data, an initial dataset with a specified ratio of event/nonevent frequency to test the HYRES methodology.

For the HYRES models, the initial dataset was sampled using the %HYRES macro to produce disjoint datasets using specified random number seeds. The data were initially separated into training and test (holdout) subsets in 75%/25% proportions. The initial training dataset (75% of observations) was further stratified by the target variable into training and validation datasets for model construction according to 75%/25% proportions. Three sets of disjoint T/V datasets were produced from the initial dataset to be consistent with the 3-fold cross-validation approach. Following the HYRES methodology, all of the Signal observations from the initial dataset were randomly assigned to T/V datasets in specified 75%/25% proportions. The Noise samples for training and validation were randomly sampled per dataset in specified 75%/25% proportions. Decision tree, logistic regression, and neural network models were built for each of the disjoint HYRES datasets extracted from the initial dataset, and all nine models (one decision tree, one regression model, and one neural network model for each of the HYRES datasets) were input into one Ensemble node, which averaged the posterior probabilities for class targets. Using one Ensemble node ensures the greatest diversity of predicted outcomes generated by the various models applied to the separate datasets.

For the 3-fold cross-validation models, the initial dataset was used to produce three disjoint stratified datasets each containing one-third of the event and one-third of the nonevent observations assigned randomly. The Data Partition node was used to produce stratified T/V datasets for subsequent modeling. Each stratified T/V dataset was used to build decision tree, logistic regression, and neural network models, and all of the models were input into one Ensemble node. The Ensemble node was used to average the posterior probabilities in the same manner as for the HYRES models.

The initial holdout dataset for each set of data (25% of the sample's observations, stratified by the target variable to the level of the event in the population) was used to score the HYRES and 3-fold cross-validation ensemble models, and the Assessment node was used to produce performance metrics for comparison. We added a cross-entropy metric, programmed using a Code node, because it is threshold-free and not influenced by imbalanced class distributions.

Figure 3 shows the Enterprise Miner Process Flow Diagram for the Credit Card Default model.

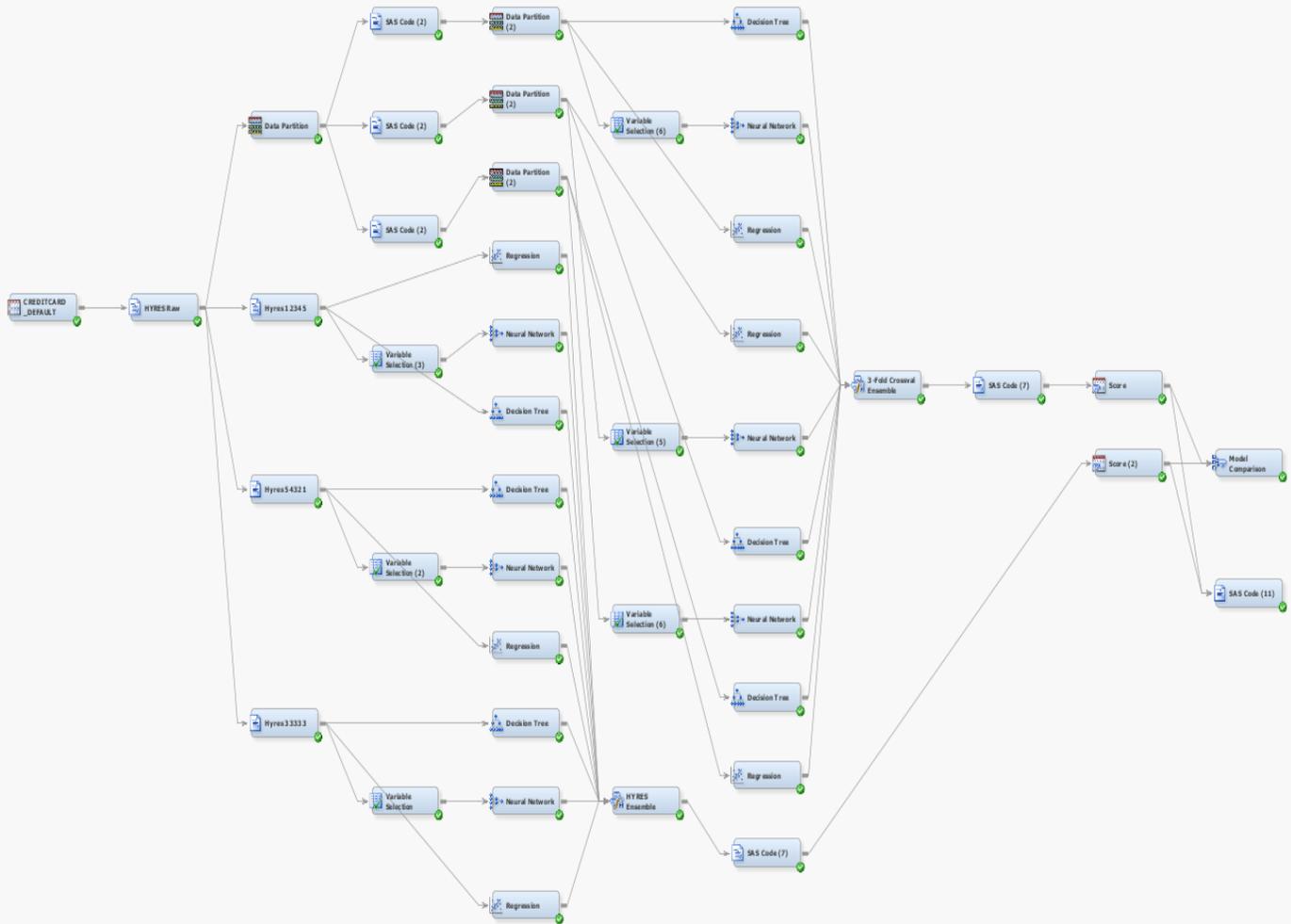


Figure 3 Enterprise Miner Process Flow Diagram for Credit Card Default Model

Figure 4 shows an excerpt of the EM PFD for the Credit Card Default HYRES 3-fold cross-validation portion of the model.

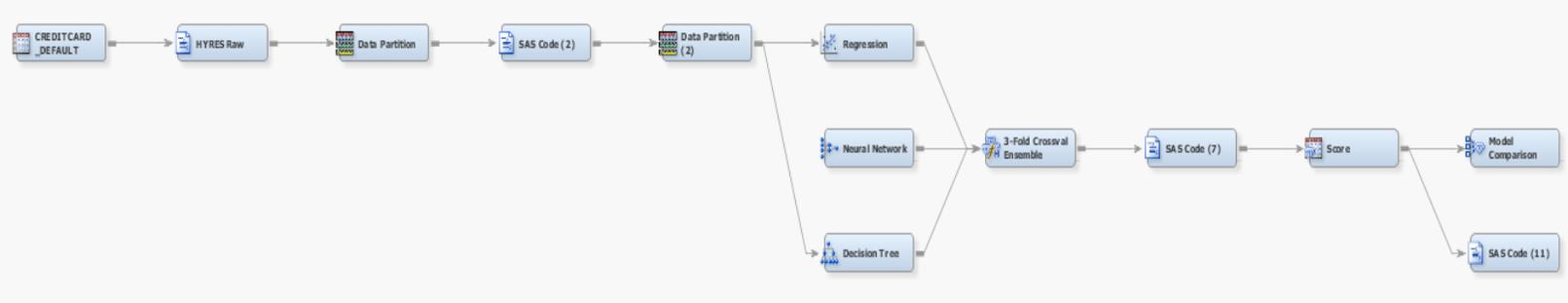


Figure 4 Nodes for one fold of 3-fold cross-validation

Figure 5 represents one branch of the nodes that implement the HYRES methodology.

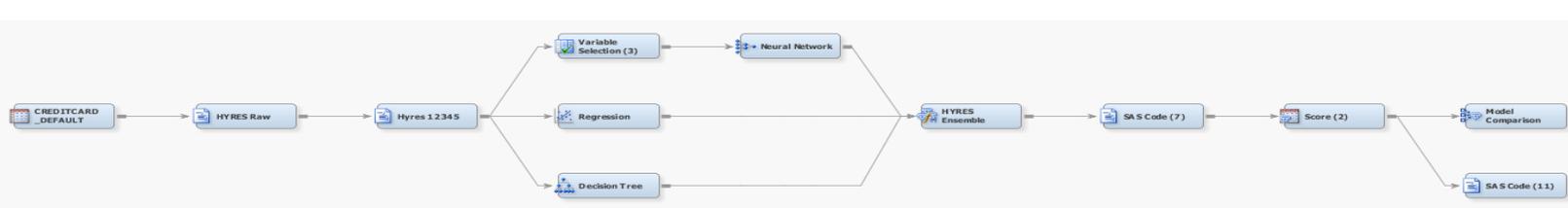


Figure 5 One branch of the nodes for the HYRES methodology

Model Performance Metrics

It is important to choose the performance metrics carefully due to the imbalanced distribution of the values of the target variable for some datasets. Reporting a result that is strongly influenced by the degree of imbalance may lead the researcher to an erroneous conclusion a model's performance⁶.

We chose the following performance metrics since they are computed over the entire set of scored observations and are not influenced by the imbalance of the distribution of events.

- ROC index

The ROC index is the area under the Receiver Operating Characteristic curve and is a measure of the optimality of the trade-off between correct classification of an observation as a true positive event compared to its being misclassified as a false positive event. See [3] for detailed description and additional references.

- Brier score

The Brier score measures the accuracy of probabilistic predictions. It is a cost function that computes the mean squared difference between predicted probabilities p_i and actual outcomes o_i . See [4] for the original definition.

- Top-decile lift

Top-decile lift is the ratio of the percent of events in the top 10% of the observations in a dataset sorted by decreasing probability of event to the average percent of events computed over all observations. See [5] for a good description and an example.

- Cross-entropy error

Cross-entropy error is a loss function that is computed by averaging all of the cross-entropies in the sample data. See [6] for a concise exposition of the cross-entropy error.

Results of HYRES and 3-Fold Cross-Validation Model Performance

We report the results of the HYRES vs 3-fold cross-validation comparison in Table 3. Performance indicators for ROC index, Brier score, and top-decile lift were taken from the Model Comparison node using the Test dataset results. The cross-entropy error function was computed in a SAS Code node.

The modeling results for HYRES and 3-fold cross-validation for the first random sample⁷ are presented in Table 3. KPIs in bold font represent the superior performance of either the HYRES approach or of 3-fold cross-validation for a particular set of modeling data.

⁶ An example of a metric that is strongly affected by the imbalance of the distribution of the values of the target variable is the misclassification rate. If, in a sample of 1,000,000 observations, there are 1,000 misclassifications the misclassification rate is clearly 0.1%, which appears to be an excellent result. However, if all of the nonevents are correctly identified as nonevents, then the misclassifications occur among the events only. If there are 2,000 events in the sample data used in modeling, the actual number of misclassifications is 50%, and our modeling ardor is considerably abated. In general, if there are n_{Events} and $n_{NonEvents}$, and if $n_{Events} \ll n_{NonEvents}$, then the misclassification rate $n_{Misclass} / (n_{Events} + n_{NonEvents}) \rightarrow 0$ as $n_{NonEvents}$ increases. So, the class imbalance distorts the actual significance of the metric, and is perhaps best not reported so as to avoid the appearance of excellent performance when in fact the opposite result may be true.

⁷ The random number seed used to select initial dataset events for the first random sample was 123456789, and the seed for nonevents was 987654321. See Appendix for details of %HYRES invocation.

Table 3 Ensemble Average Posterior Probability, First Random Sample

Performance Metrics									
		3-Fold Cross-Validation				HYRES			
Model	Events as % of Sample	ROC Index	Brier Score	Top Decile Lift	Cross Entropy Error	ROC Index	Brier Score	Top Decile Lift	Cross Entropy Error
Higgs Boson	0.0113	0.730	0.000112	3.403	0.001111	0.741	0.000112	3.542	0.001096
PaySim Fraud	0.1291	0.994	0.000544	9.957	0.002863	0.995	0.000540	9.962	0.002785
Credit Card Fraud	0.1727	0.975	0.000526	9.394	0.003171	0.977	0.000538	9.167	0.003167
Sales Fraud	0.3098	0.892	0.002667	7.785	0.014877	0.864	0.002690	6.817	0.015377
Adult Income	0.4954	0.878	0.005005	6.411	0.026847	0.891	0.000495	6.600	0.026509
Coverttype	3.5300	0.992	0.010393	9.668	0.039196	0.991	0.010409	9.635	0.039540
Wilt	5.3937	0.998	0.008255	9.976	0.032897	0.998	0.008208	9.976	0.034348
Bank Direct Mktg	11.2654	0.806	0.079261	4.451	0.277790	0.802	0.079257	4.509	0.278370
Cred Card Default	22.1200	0.774	0.135739	3.234	0.434150	0.772	0.136036	3.204	0.434670

We observe that the HYRES methodology does not produce consistently superior results compared to 3-fold cross-validation. For many KPI's, there is very little difference between the HYRES results and 3-fold cross-validation results.

Equivalence of Information Contents of Noise Samples

To ensure the uniformity of the Noise samples being created, the modeler must verify that the information content of each of the Noise samples is consistent with the others. One possible approach to verifying the uniformity of the Noise samples would be to create different random samples of the Noise data and build several models, holding the Signal data constant and varying the Noise data. If the results of the several models were similar within reasonable variability, we would be justified to conclude that the Noise samples had consistent information content.

The experiment was repeated with a second random sample⁸ to verify the consistency of the information content of the Noise samples, as described *supra*. The modeling results for HYRES and 3-fold cross-validation for the second random sample are presented in Table 4.

Table 4 Ensemble Average Posterior Probability, Second Random Sample

Performance Metrics									
Model	Events as % of Sample	3-Fold Cross-Validation				HYRES			
		ROC Index	Brier Score	Top Decile Lift	Cross Entropy Error	ROC Index	Brier Score	Top Decile Lift	Cross Entropy Error
Higgs Boson	0.0113	0.693	0.000102	2.977	0.001021	0.717	0.000102	3.817	0.001009
PaySim Fraud	0.1291	0.996	0.000527	9.972	0.002731	0.992	0.000553	9.881	0.003072
Credit Card Fraud	0.1727	0.976	0.00061	9.138	0.003659	0.978	0.00057	9.309	0.00355
Sales Fraud	0.3098	0.861	0.00291	7.275	0.016087	0.845	0.002937	6.731	0.016484
Adult Income	0.4954	0.878	0.004899	6.04	0.02554	0.876	0.00499	6.249	0.02681
Coverttype	3.5300	0.991	0.010374	9.689	0.039995	0.991	0.010715	9.646	0.039416
Wilt	5.3937	0.984	0.014846	9.733	0.055858	0.986	0.013734	9.733	0.053238
Bank Direct Mktg	11.2654	0.805	0.076735	4.357	0.27022	0.81	0.076225	4.48	0.26847
Cred Card Default	22.1200	0.782	0.13384	3.245	0.42892	0.783	0.133422	3.263	0.42778

We see that, while there are reasonable differences in the results, the ranges of values of the performance metrics are essentially unchanged, so we may conclude that the information content of the Noise samples is consistent.

We observe the same pattern of performance as for the first random sample, i.e., the HYRES methodology does not produce consistently superior results compared to 3-fold cross-validation.

Conclusion

We investigated the hypothesis that, for selected populations containing rare events, a hybrid sampling approach designed to improve the signal-to-noise ratio of the event population (S) to the nonevent population (N) would produce superior classification performance compared to standard cross-validation techniques.

We developed a SAS macro, %HYRES, to perform stratified sampling of a dataset prior to applying the HYRES and 3-fold cross-validation methodologies to model-building in the context of SAS Enterprise Miner.

We built classification models for nine datasets characterized by event frequencies from abundant to rare (event rates ranged from 22% to <0.05%) and numbers of observations from large to small (millions to thousands), and chose key performance metrics that would not be influenced by the class imbalance due to the sparsity of an event.

We observed that there was no consistent pattern of improvement in classification performance due to applying the HYRES methodology compared to 3-fold cross-validation, and that the results of both approaches were very closely matched. We conclude that the HYRES methodology does not represent a systematic improvement over conventional cross-validation methodology.

⁸ The random number seed used to select initial dataset events for the second random sample was 987654321, and the seed for nonevents was 123456789.

Appendix

%HYRES Macro

We wrote the %HYRES (hybrid rare event sampling) SAS macro to produce the T/V/T datasets containing specified percentages of signal and noise (S_T , S_V , S_H , N_{Tn}), for our experiments.

The macro has the following parameters:

```
%macro HYRES( DSNIN          /* name of input  dataset          */
              , DSNOUT        /* name of output dataset        */
              , TARGET_VAR    /* name of target variable       */
              , EVENT_VAL     /* value of event                 */
              , LEVEL_EVENT_PCT= /* % of events in data to include in
                               sample using level-based sampling */
              , N_EVENT=0     /* number of events in &DSNOUT   */
              , N_NONEVENT=0  /* number of nonevents in &DSNOUT */
              , PCT_EVENT=100 /* percent of events in &DSNOUT  */
              , PCT_NONEVENT=50 /* percent of nonevents in &DSNOUT */
              , PCT_TRN=50    /* % of obs to be labelled 'TRN' */
              , PCT_VAL=25    /* % of obs to be labelled 'VAL' */
              , PCT_TST=25    /* % of obs to be labelled 'TST' */
              , SAMPLE_PCT=   /* % of sample pop to comprise sample
                               data using level-based sampling */
              , SEED_EVENT=12345 /* random number seed for event */
              , SEED_NONEV=54321 /* random number seed for nonevent */
              ) ;
```

As an example of use, let us construct a modeling dataset using %HYRES. We define it to have 1,000,000 observations with target class percentages of 1% for the event value 1 and 99% for the value 0. We want the train/validate/test datasets to have 50%/30%/20% for both the event and nonevent observations. We also want to use only 50% of the target events since there are such a surplus of them. Then we write the following code:

```
data test ;
  do i = 1 to 1000000 ;
    tgt_val = mod( i, 100 ) = 0 ;
    output ;
  end ;
  drop i ;
run ;
```

We have created a modeling dataset containing 10,000 observations with target value 1 and 990,000 observations with target value 0. The macro invocation shown below will produce an output dataset containing 50% of the target values for events and nonevents in train/validate/test proportions 50%/30%/20%.

```
%HYRES( test
        , test_out
        , tgt_val
        , 1
        , pct_event=50, pct_nonevent=50, pct_trn=50, pct_val=30, pct_tst=20
        )
```

and we observe the following result from PROC FREQ in Table 5:

Table 5 Hybrid Rare-Event Sampling, 1% Event Rate

Hybrid Rare-Event Sampling					
The FREQ Procedure					
Frequency Percent Row Pct Col Pct	Table of tgt_val by _dataset_class_				
	tgt_val(Target Variable)	_dataset_class_(Dataset Class)			
		TRN	TST	VAL	Total
0	247268 49.45 49.95 99.01	99038 19.81 20.01 99.01	148694 29.74 30.04 98.97	495000 99.00	
1	2472 0.49 49.44 0.99	986 0.20 19.72 0.99	1542 0.31 30.84 1.03	5000 1.00	
Total	249740 49.95	100024 20.00	150236 30.05	500000 100.00	

The percentages allocated to event and nonevent are 1% and 99%, as in the original dataset, and the percentages allocated to train/validate/test categories are 50%/30%/20%, as specified. All that remains now is to build a model based on the data.

The T/V/T datasets may be created using the following SAS statements:

```
data train validate test ;
  set test_out ;

  if _dataset_class_ = 'TRN' then output train    ; else
  if _dataset_class_ = 'VAL' then output validate ; else
                                     output test    ;

  drop _dataset_class_ ;
run ;
```

If we had desired to use all of the event data, we would have specified `pct_event=100` and we would observe the results in Table 6:

Table 6: HYRES Sampling, 100% of Events

Hybrid Rare-Event Sampling

The FREQ Procedure

Frequency Percent Row Pct Col Pct	Table of tgt_val by _dataset_class_				
	tgt_val(Target Variable)	_dataset_class_(Dataset Class)			
		TRN	TST	VAL	Total
0	247896	98914	148190	495000	
	49.09	19.59	29.34	98.02	
	50.08	19.98	29.94		
	98.05	97.93	98.03		
1	4933	2092	2975	10000	
	0.98	0.41	0.59	1.98	
	49.33	20.92	29.75		
	1.95	2.07	1.97		
Total	252829	101006	151165	505000	
	50.07	20.00	29.93	100.00	

In Table 6, we see that all of the events have been distributed to the train/validate/test categories in the desired proportions so that the nonevent/event ratio of 99:1 in Table 5 has been halved in Table 6. Significantly, the proportion of events in the train/validate/test datasets has doubled since all of the events are now available for modeling. Standard cross-validation techniques would reduce the number of events and might not capture enough behavioral information in the case of rare event frequencies.

References

- [1] Hastie, T, Tibshirani, R, Friedman, J. (2008), The Elements of Statistical Learning, Second Edition, Springer, New York.
- [2] James, G., Witten, D., Hastie, T., Tibshirani, R. (2014). An Introduction to Statistical Learning, Springer, New York.
- [3] https://en.wikipedia.org/wiki/Receiver_operating_characteristic
- [4] Brier, G. W. (1950). "Verification of forecasts expressed in terms of probability", *Monthly Weather Review*, 78(1):1-3.
- [5] <http://www.dmstat1.com/res/DecileAnalysisPrimer.html>
- [6] https://en.wikipedia.org/wiki/Cross_entropy (see the section entitled "Cross-entropy error function and logistic regression")

Data Sources

- [1] Kohavi, Ronny, and Barry Becker. (1996). UCI Machine Learning Repository [\[http://archive.ics.uci.edu/ml\]](http://archive.ics.uci.edu/ml). Irvine, CA: University of California, School of Information and Computer Science.
- [2] Moro, S., Cortez, P., Rita, P. (2014) "A Data-Driven Approach to Predict the Success of Bank Telemarketing", *Decision Support Systems*, Elsevier, 62:22-31.
- [3] Blackard, Jock A., and Denis J. Dean. (1998). UCI Machine Learning Repository [\[http://archive.ics.uci.edu/ml\]](http://archive.ics.uci.edu/ml). Irvine, CA: University of California, School of Information and Computer Science.
- [4] Yeh, I. C., and Lien, C. H. (2009). "The Comparisons of Data Mining Techniques for the Predictive Accuracy of Probability of Default of Credit Card Clients". *Expert Systems with Applications*, 32(2), 2473-2480.
- [5] Dal Pozzolo, Andrea, Caelen, Olivier, Johnson, Reid A., Bontempi, Gianluca. (2015). "Calibrating Probability with Undersampling for Unbalanced Classification" in "Symposium on Computational Intelligence and Data Mining (CIDM), IEEE, 2015. (<https://www.kaggle.com/dalpozz/creditcardfraud>)
- [6] Baldi, P., P. Sadowski, and D. Whiteson. (July 2, 2014). "Searching for Exotic Particles in High-energy Physics with Deep Learning." *Nature Communications* 5.
- [7] Lopez-Roja, E. A., Elmir, A., Axelsson, S. (2016). "PaySim: A Financial Mobile Money Simulator for Fraud Detection". The 28th European Modeling and Simulation Symposium-EMSS, Larnaca, Cyprus.
- [8] Torgo, Luis. (2011). *Data Mining With R: Learning with Case Studies*. Chapman & Hall/CRC Press, Boca Raton, FL. <http://www.dcc.fc.up.pt/~ltorgo/DataMiningWithR/>
- [9] Johnson, B., Tateishi, R., Hoan, N. (2013). "A hybrid pansharpening approach and multiscale object-based image analysis for mapping diseased pine and oak trees". *International Journal of Remote Sensing*, 34 (20), 6969-6982.

Acknowledgements

We thank John Amrhein, Leonardo Auslender, Mark Leventhal, Joseph Naraguma, and Doug Wielenga who reviewed preliminary versions of this paper and contributed their helpful comments.

Contact Information

Your comments and questions are valued and encouraged. Contact the author at:

Name: Ross Bettinger

E-mail: rsbettinger@gmail.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.