Print

Summary of Lesson 4: Measuring Model Performance

Honest Assessment

After you create a family of increasingly complex models, you need to compare and evaluate them on the training and validation data. You can use a variety of metrics to measure model performance in terms of fit versus complexity. The fit statistics tend to increase as complexity increases. Some of this increase happens because the model is capturing relevant trends in the data. However, some of the increase is due to overfitting. The difference between the training fit line and the validation fit line, known as shrinkage, is another statistic that some modelers use when measuring a model's overall predictive power. So, when comparing models, you might use a rule that says "Choose the simplest model that has the highest validation fit measure, with no more than 10% shrinkage from the training to the validation results."

When the target event is rare, you might not be able to afford to split your data because you want to use all of the target event cases to fit the model. In this situation, you can use other honest assessment approaches, such as bootstrapping and k-fold cross-validation. Bootstrapping is repeated sampling with replacement. In k-fold cross-validation, you split your data into *k* parts, also called folds. The benefit of using k-fold cross-validation is that you use all of the data for both training and validation.

View the "Preparing the Validation Data" demonstration that shows how to use SAS to prepare the validation data.

Common Metrics for Model Performance

Typically, you use predictive models to categorize cases using a cutoff. For a given cutoff, you need to assess how well your predictive model performs. The fundamental assessment tool for model performance is the confusion matrix. The confusion matrix is simply a cross-tabulation of predicted classes and actual classes. You can use the confusion matrix to calculate statistics that measure model performance. Some of the most common statistics are accuracy, error rate, sensitivity, positive predicted value, specificity, and negative predicted value.

You can measure model performance across all cutoffs by using the receiver operating characteristic curve, also called the ROC curve. The ROC curve displays the sensitivity (also known as the true positive rate) and 1 minus specificity (also known as the false positive rate) for the entire range of cutoffs. On the ROC curve graph, sensitivity is on the Y axis, and 1 minus specificity is on the X axis. As the cutoff decreases, more cases are allocated to class 1, the sensitivity increases, and the specificity decreases. As the cutoff increases, more cases are allocated to class 0, the sensitivity decreases, and the specificity increases.

One widely used measure of model performance is the gains chart. The cumulative gains chart displays the positive predicted value on the Y axis and the depth on the X axis. Depth equals the total percentage of cases that are allocated to class 1. As the cutoff increases, the depth decreases. The marginal rate equals the proportion of events in the sample adjusted to the true population event rate, such as the rate of response to a promotion. A lift chart is a variation of the gains chart. The lift equals the positive predicted value divided by the marginal rate. Just like with a regular gains chart, a lift chart for a model with good predictive power has a curve shaped like a steep ski slope.

If you create a validation data set by splitting oversampled data, then the validation data is also a biased sample. Oversampled data affects some performance measures. Although oversampling does not affect sensitivity or specificity measures, it does affect positive predicted values and negative predicted values. Because gains charts and lift charts rely on positive predicted values, they are also

affected by oversampling. Before you create gains charts and lift charts, you need to adjust the confusion matrix for oversampling.

If you oversampled, you need to adjust your confusion matrix so that it matches your population. To do this, you need to know the values for π_1 and π_0. You also need to know the values for sensitivity (represented by *Se*) and specificity (represented by *Sp*).

View the "Measuring Model Performance Based on Commonly-Used Metrics" demonstration that shows how to use SAS to measure model performance based on commonly-used metrics.

Profit-Based Metrics

Different cutoffs produce different classification allocations and different confusion matrices. Higher cutoffs decrease sensitivity and increase specificity. Lower cutoffs decrease specificity and increase sensitivity.

In business, you typically choose cutoffs based on profit, rather than sensitivity or specificity. The optimal cutoff maximizes the total expected profit. To choose the optimal cutoff, you generate a profit matrix. The profit matrix displays the expected profit for each true negative, false negative, false positive, and true positive. Profit equals revenue minus cost.

You can use the profit matrix to calculate the total expected profit for a cutoff. To maximize profits, you need to find the decision point that has the highest expected profit across all cutoffs. You use Bayes' rule to do this.

You must have the cost and profit information for your business problem to calculate Bayes' rule. In many situations, gathering profit information can be difficult. When this is the case, many business analysts use a cutoff of p_1. The central cutoff of π_1 tends to maximize the mean of sensitivity and specificity. Because increasing sensitivity usually corresponds to decreasing specificity, the central cutoff tends to equalize sensitivity and specificity.

If you have a profit matrix, you can use profit to assess fit by generating an empirical profit plot. To do this, you create a plot in which average profit is on the Y axis and depth is on the X axis. A profit plot is a simple way to illustrate how deep to go into your sample to maximize profits.

When your sample is not representative, a technique to adjust the data is to compute sampling weights. When a rare target event is oversampled, class 0 is under-represented in the sample. Consequently, a class-0 case should actually count more in the analysis than a class-1 case. The sampling weights adjust the number of cases in the sample so that the classes are now in the same proportion in the adjusted sample as they are in the population.

View the "Using a Profit Matrix to Measure Model Performance" demonstration in SAS that shows how to use a profit matrix to measure model performance.

Kolmogorov-Smirnov Statistic

In order to assess the overall predictive power of a model, you need to look at how well the model discriminates between events and non-events. You can use a class separation graph to do this. The lines on the graph represent the distributions for class 0 (the actual non-events) and class 1 (the actual events). When the distributions overlap a lot, it means that the model does a poor job of discriminating between events and non-events. The more the distributions overlap, the weaker the model is.

The Kolmogorov-Smirnov two-sample test is commonly used to assess how well a model distinguishes between events and non-events. The Kolmogorov-Smirnov test produces a value called the K-S statistic. The K-S statistic that we are interested in is called the D statistic in the NPAR1WAY procedure. The higher the value of D is, the better the model distinguishes between events and non-events. The K-S test is concerned with the shape, variance, and central tendency of a distribution. However, when you assess the predictive power of a model, differences in central tendency are more important than differences in shape and variance. The Wilcoxon-Mann-Whitney test statistic has more

power than the K-S statistic in its ability to measure differences in central tendency. The value of the Wilcoxon-Mann-Whitney test statistic is equivalent to that of the c statistic.

View the "Using the K-S Statistic to Measure Model Performance" demonstration that shows how to use the K-S statistic to measure model performance.

Model Selection Plots

The modern strategy for predictive modeling is to create a family of increasingly complex predictive models and choose the one that generalizes the best. You can do this by using some custom macros.

The ROC curve is a measure of a model's predictive accuracy. To create and compare ROC curves for multiple models, you use both the ROC and ROCCONTRAST statements in PROC LOGISTIC. You use one ROC statement for each of the models that you want to compare. You can use a label to identify the output for each ROC statement. The specification in each statement specifies the models to be compared. The specification can be either a list of input variables that have previously been specified in the MODEL statement, or PRED=variable, where the variable does not have to be specified in the MODEL statement such as the predicted probability. The PRED= option enables you to input a criterion produced outside PROC LOGISTIC. The ROCCONTRAST statement compares the ROC curves for the models that you specified in the ROC statements. You can specify only one ROCCONTRAST statement. You specify a label to identify the contrast statistics in the output. You provide a contrast specification to specify how the models will be compared. If no contrast is specified, then by default SAS produces a contrast matrix of the differences between each ROC curve and a reference curve.

ROC *<'label'> <specification> </ option(s)>*;
ROCCONTRAST *<'label'> <contrast> </ option(s)>*;

View the "Comparing ROC Curves to Measure Model Performance" demonstration that shows how to compare ROC curves to measure model performance.

The process of selecting inputs for a model, fitting that model, and evaluating that model's fit on the validation data set can be time-consuming. However, you can automate the process with macro programming. The Assess and Fitandscore macros enable you to consider many candidate models in a small time frame. These two macros take a series of models generated by the best-subsets logistic regression and compare them on the validation data performance. You can generate plots of the results, which show the performance gains as a function of model complexity. These plots can be a helpful tool as you make your final model selection.

View the "Comparing and Evaluating Many Models" demonstration that shows how to use SAS to compare and evaluate many models.

Close