No. 1, 2018

RISK INSIGHTS®

# Classification Model Performance

*by Louis Rossouw, Gen Re, Cape Town*

Insurers are increasingly developing prediction models to use in their insurance processes. Often these models are using traditional techniques, but more and more we see machine learning techniques being applied.

In practice these techniques are applied to underwriting cases, policy applications or even claims. An example might be a model that is being used to predict which cases will be assessed as "standard" before an underwriter sees it. This is called a "classification model", and is used to classify data points into discrete buckets (yes or no, standard or not, etc.).

The modelling techniques used in such applications can be quite simple or very complex. Examples of these techniques include:

- Logistic Regression (typically a generalised linear model – GLM)
- Decision Trees
- Random Forrest
- Support Vector Machines
- Gradient Boosting Techniques
- Neural Networks

Traditional techniques, such as regression-based models, produce models that are human-readable. One can clearly see the impact of each variable in the model on the outcome. However, many of the machine learning techniques produce models that are not as easily understood by looking at them directly. They produce output but the inner workings are hidden or are too complex to fully understand. These are the so-called "black-box" models.

With such a wide range of choice in models, how do we assess the accuracy and quality of these models to determine which is the best to use? How do we consider the performance of models that are easy to understand, compared to various black-box models? How do we assess the relative value to the insurer of these models?

## Contents



## About This Newsletter

*Risk Insights* is a technical publication produced by Gen Re for life and health insurance executives worldwide. Articles focus on actuarial, underwriting, claims, medical and risk management issues. Products receiving emphasis include life, health, disability income, long term care and critical illness insurance.

For the rest of this article, we will focus on a simple binary classifier predicting whether a particular underwriting application should be considered "standard" (no loadings or underwriting conditions) or not (declined, with a loading or other underwriting terms applied). The potential use of such a model is to bypass traditional underwriting to save time and cost for a subset of cases.

## Training & testing data

When creating a classifier model, one would typically have a dataset with historic cases and the recorded outcome of those cases. In our underwriting example, this might be data related to the applicant and the recorded underwriting decision (standard or not).

It is advisable to split the past data into at least two categories:

- Training data that will be used to fit the model(s) in question; i.e. this data is used to "train" the model
- Testing data that will be used to evaluate the model(s) to determine how good the model(s) perform

Often a validation data set is also used to refine modelling parameters before the model is tested.

The main reason for the separation between testing and training is to ensure that the model performs well with data on which it has not been trained. In particular this identifies the problem of over-fitting, which happens when a particular model seems to predict too accurately using the training data. However, when this model is then checked against other testing data, the performance of the model degrades significantly. One can then say that the model is over-fitting the training data.

Typically 10% to 30% of data is held back as testing data. This would depend on the overall availability of data and the degree to which models could potentially over-fit. One could select testing data as a random subset of your data or, for example, base the selection of data on time (e.g. by holding out the latest year of data as a testing subset).

## The confusion matrix

To assess the quality of a binary classifier, we can generate a confusion matrix using our testing data. On this data we would run the model to produce predicted outcomes, and also have the actual outcomes in the data. We would tabulate a matrix with the counts of cases where the model identified the outcome as positive (standard in our example) and the actual outcome was positive (standard also). This is the "True Positive" count. Similarly, we count the number of cases where the model correctly predicted the negative as the "True Negatives". We also count the errors where the model predicted negative but the result was positive and vice versa. This generates a confusion matrix as shown:

| | | Predicted | |
|---|---|---|---|
| Actual | | Positive | Negative |
| | Positive | True Positives | False Negatives |
| | Negative | False Positives | True Negatives |

In a confusion matrix, false positives are also known as Type I errors and false negatives are known as Type II errors.

One can then use this confusion matrix to classify the model using various metrics:

$$Accuracy = \frac{True\ Positives + True\ Negatives}{Total\ Cases}$$

$$Sensitivity = \frac{True\ Positives}{True\ Positives + False\ Negatives} = \frac{True\ Positives}{Actual\ Positives}$$

$$Specificity = \frac{True\ Negatives}{True\ Negatives + False\ Positives} = \frac{True\ Negatives}{Actual\ Negatives}$$

The accuracy measure is an overall measure of accuracy. Sensitivity indicates how many of the positives are actually identified as positive. The interplay of these variables indicates how good a model is. For example, if we have two models predicting the outcomes of whether cases are standard or not, we can tabulate a confusion matrix for each.

Model A is a very simple (and very inaccurate) model that predicts that every single case will be standard.

| Model A | | Predicted | |
|---|---|---|---|
| | | Standard | Non-Standard |
| Actual | Standard | 80 | 0 |
| | Non-Standard | 20 | 0 |

From the confusion matrix we can assess that there were 100 cases. Our model predicts all cases to be standard. Our sensitivity is then 100% and

our accuracy looks good at 80%. However, our specificity is poor at 0%, which indicates a poor-performing model.

Clearly, one needs to consider multiple measures to understand how good a model is. A more realistic model might look something like this (on the same data):

| Model B | | Predicted | |
|---|---|---|---|
| | | Standard | Non-Standard |
| Actual | Standard | 61 | 19 |
| | Non-Standard | 8 | 12 |

In this case, the overall accuracy is 73%; sensitivity is 76.25%, and specificity is 60%. This appears to be a reasonable model.

## Model scores & threshold

Most classifier models don't only produce a binary classification as output. They typically produce a score to classify cases as positive or negative. The score is typically converted as a percentage. This score does not always necessarily imply a true probability, especially for the machine learning techniques, and often only indicates a ranking of cases, rather than a strict probability.

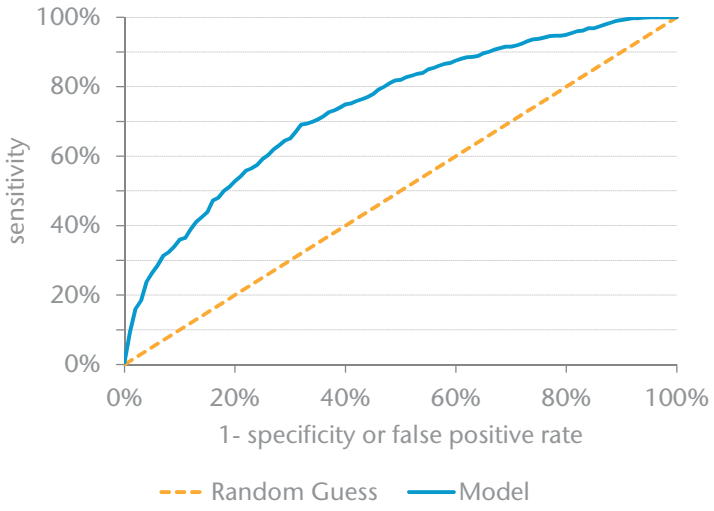
Given that each case would have a score, one would need to assign a threshold (or cut-off) at which point the model outcome could be considered as a positive. For example, a model might produce various scores for various cases, and with a threshold of 80%, would treat a case as a standard case only when the score is above 80%.

Each of these thresholds then implies a specific confusion matrix. Thus, with a threshold of 0%, we would end up in the situation of model A shown above; i.e. we will predict every case as standard, with the sensitivity of 100% but a specificity of 0%. Similarly, a threshold of 100% results in all cases being classified as non-standard, with 0% sensitivity and 100% specificity.

## Receiver Operating Characteristic Curve (ROC)

Changing the threshold for a particular model over all values between 0% and 100% allows one to plot a curve of the various specificity and sensitivity values. The Receiver Operating Characteristic (ROC) plots *sensitivity* (y-axis) vs. *specificity* (x-axis) for every value of the threshold. Note that 1–*specificity* is also the false positive rate. Figure 1 is an example of such a ROC curve.

Figure 1 – Received Operating Characteristic Curve




On the bottom left, we see the case where the threshold is 100%. This is equivalent to our model for predicting all cases as non-standard (sensitivity is 0% but specificity is 100%, or the false positive rate is 0%). On the top right we have the situation where all cases are predicted as being standard (sensitivity is 100% but specificity is 0%, or the false positive rate is 100%).

The diagonal dashed line represents the expected outcome for a model randomly assigning scores to cases. The blue line represents the outcomes for a particular model that is better than random guessing. The overall quality of the model could be determined by assessing the area under the blue curve. In this case, it's calculated at 74%, which represents a reasonably good model.

We can pick two random cases – one from the actual standard cases and one from the actual non-standard cases. It has been shown that the area under the curve (74% in this example) is equivalent to the probability that the standard case have will have a higher score than the non-standard case. Thus this area provides an overall measure at how good the model is at sorting standard cases from non-standard cases.

The random guessing model has an area under the curve of 50%, so that is the minimum requirement upon which to improve. A perfect model would have an area under the curve of one, and the model would produce a point in the top left corner of Figure 1.

Generally models with area under the curve of between 50% and 60% are considered unsuccessful, and models exceeding 90% as very good. A result in the 70%-80% range is considered fair to good. Note that a model with an area under the curve of less than 50% should in fact be reversed, as it is in effect predicting the opposite outcome more successfully; generally the area under the curve varies between 50% and 100%.

## Gini coefficient

The Gini coefficient also relates to assessing classifier models. It is actually directly related to the area under the ROC curve mentioned above. The Gini coefficient is calculated from the area under the curve (AUC) as $2AUC-1$. A 74% area under the curve becomes a Gini coefficient of 48%, which is fair. The Gini coefficient hence effectively ranges between 0% and 100%, though it can be negative, in which case the model should really be reversed as in the case of an area under the curve of less than 50%.

## Comparing models

Different models and modelling techniques will result in different performance on different data and different problem scenarios. To compare two different models – and select the better one – we can compare each area under the ROC curve (or Gini coefficients). We can say that a model A is better if its area under the curve is bigger than B.

However, there are some qualifications. If we plot the models and A's ROC crosses B's ROC curve, we cannot say that A is always better. At some sensitivity and specificity levels, B may well be better. However, if A's ROC curve does not cross B's and its area under the curve is bigger, that means that A is essentially always better than B.

For models that are close in terms of their ROC curves and the areas under the curves, practical considerations – such as implementation details – may hold sway.

## Ensemble models

There are various ways to improve models dependant on the technique involved. One interesting technique to consider is constructing ensemble models. Once we have multiple models that produce a score for a particular outcome, we can start combining them in interesting ways to produce ensemble scores. These can be used to improve the area under the curve for these models even further.

Take, for example, a Random Forrest classifier and a logistic regression model, both predicting standard risks. A new score can be calculated as the average of these two classifiers and then assess it as a further model. Usually the area under the curve improves for these ensemble models.

We could also build a model of models by fitting a combination logistic regression model to the various underlying models modelling the final outcome, essentially using the data to suggest how the various models should be weighted. Usually a further validation sample of data (not used for training nor for testing) would be retained to produce this type of ensemble mode.

## Business optimisation

Given a model with a particular ROC curve, we can now decide how to apply this model in practice. Given that each error type (false-positives and false-negatives) would have associated costs and benefits to the business, we can then estimate the point at which the cost is minimised or the benefit is maximised, thus setting the best threshold for each model. From those outcomes, we can compare the best business performance of various models to make a final decision on a particular model.

In the example of modelling standard underwriting decisions, we can consider the implications of each category of outcome on the present value of profits per application:

- True positives (cases correctly classified by the model as standard) may see an increase in value per policy as we would see higher placements from this category due to their having lower per policy selling expenses (higher conversion ratio due a customer-friendly streamlined process) and lower medical and underwriting costs.
- False positives (cases incorrectly classified as standard) may face problems of increased claims relative to premiums. There is also a risk of anti-selection here if applicants understand how to influence their scores.
- False negatives (cases that are incorrectly classified as non-standard) may be very similar to the current process, so we may need to use

a current conversion ratio and take-up rates in assessing the present value of profits per application (assuming the underwriting follows our current approach for these).

- True negatives (cases correctly classified as negative) may again be fairly consistent to our current treatment of non-standard cases.

Given values for each of the above, it's possible to estimate an optimal threshold for use in our model, one that will bring maximum value per application to the business. This would correspond to a single point on the ROC curve. The selected outcome should be tested for sensitivity to assumptions as many of the assumptions made in the determining the value of the various scenarios might be subjective.

If we have multiple models (where the ROC curves crossed as before), or a model for which we could not calculate the ROC curve, we could then compare their best value produced from the various models to determine the best model. When comparing these values, one would need to make sure the cost of running the model is included. Some models might have cost implications, either based on the data they used or due to technical implementation issues.

## Conclusion

This article has outlined approaches to assessing the performance of models used in classification problems.

We have made these clarifications:

- It's necessary to have hold-out data samples for testing purposes.
- It's possible to compare these models' performance using measures in a confusion matrix.
- One can estimate numbers, such as specificity and sensitivity, from these confusion matrices.
- One could also look more generally at using the ROC curve and the area under the curve to obtain a general sense of the quality of the model.
- Ensemble techniques can be used to combine scores from various models to produce better models.
- Once we apply expected business values for various outcomes of the model, it can be easier to decide what trade-off between sensitivity and specificity would be the best value for the business.
- There may be practical considerations to take into account, including subjective assumptions in valuing various outcomes and technical implementation details that may result in different outcomes.
- Changing the process may result in changed behaviours, which will invalidate the modelling. Behavior that is particularly anti-selective needs to be assessed in this context.

This relatively simple overview of the field gives a sense of how to assess these models objectively, and how to assess the value of the model for the business taking into account the performance of the model.

## References


James, G., Witten, D., Hasties, T.& Tibshirani, R. (2013). An Introduction to Statistical Learning. Springer.

Matloff, N. (2017). Statistical Regression and Classification: From Linear Models to Machine Learning. CRC Press.

Fawcett, T. (2006) An introduction to ROC analysis. Pattern Recognition Letters 27, 861–874.


## About the Author


**Louis Rossouw** *is a Research and Analytics Actuary in Gen Re's Cape Town office, supporting South Africa and the UK. Louis joined Gen Re in 2001 and has previously worked on individual pricing and product development, group pricing and reserving. He also spent two years in Gen Re's Singapore branch as Regional Chief Actuary. Louis can be reached at Tel. +27 21 412 7712 or lrossouw@genre.com.*


*The difference is...the quality of the promise®*

genre.com | genre.com/perspective | Twitter: @Gen_Re

**General Reinsurance AG**
Theodor-Heuss-Ring 11
50668 Cologne, Germany
Tel. +49 221 9738 0
Fax +49 221 9738 494

*Editors:*
*Ulrich Pasdika, ulrich.pasdika@genre.com*
*Ross Campbell, ross_campbell@genre.com*