Article from

CompAct

April 2019

Issue 59

Deep Learning and Actuarial Experience Analysis

By Kevin Kuo, Bob Crompton and Frankie Logan

Deep learning is a type of artificial intelligence that has been successfully applied in areas that involve large amounts of data and have nonlinear relationships between the inputs and outputs. Perhaps the two most widely known areas are image recognition and gameplay. Deep learning has not typically been used in areas with small or medium-sized data sets or in areas where there are strong linear relationships between input and output. Deep learning does not usually provide as much added value to these areas as it does to perceptual tasks with data-intensive nonlinearities.

For this reason, deep learning is not typically a candidate for implementation in standard actuarial work. Much of actuarial work involves linear relationships and small or medium-sized data sets. In addition, much of standard actuarial work is based on robust procedures created from decades of experience. This is certainly true of experience analysis.

However, we wanted to see if it was feasible to implement some desktop version of deep learning for experience analysis. Specifically, we were interested in these parameters:

- Accuracy and consistency of deep learning results compared with standard methods.
- Level of effort in implementation and training.
- Ability to apply deep learning to related and ancillary issues arising from experience analysis.

We have applied deep learning to the 2015 Society of Actuaries (SOA) report on the lapse and mortality experience of post-level premium period term plans (SOA Report). We used the data supplied in the SOA Report. This data is grouped rather than granular at the policy level. We applied our deep learning algorithms against this grouped data yet still obtained results that were surprisingly close to the published data.

A FEW WORDS ABOUT DEEP LEARNING

"Deep learning" is the name of a particular type of artificial intelligence. This name should not be understood to mean that it always generates profound insights. Instead, deep learning refers to neural networks with multiple hidden layers as opposed to a single hidden layer. We define deep learning as follows:

> Deep learning is a statistical technique for classifying patterns, based on sample data, using neural networks with multiple layers.

Case Study

To better understand how deep learning can be applied in the insurance industry, we perform a case study by exploring how it can help us in better understanding and predicting lapse experience to improve risk management and customer retention. In particular, we study shock lapse, which is a phenomenon where insurance companies experience a higher lapse rate post-level premium term. With the increased lapse rate, a book of business can become less profitable as inflow of premium decreases and policyholders that stay with the policies are ones that "need" the coverage. The source code for the experiments is open source and available online.[1]

Data

To create a neural net model, we utilize the publicly available data from the *2014 Post Level Term Lapse & Mortality Report* published by SOA. The data comprise in-force and terminated level term policies from the participating companies. Each row of the data represents a policy block with a unique set of characteristics. For a more detailed description of the underlying data, please refer to the SOA Report. We used policy year 2010 to split out the training (policy year < 2010), validation (policy year = 2010) and testing (policy year > 2010) sets to build our model. The training and validation sets are used to fit and assess candidate models for hyperparameter tuning, while the test set is reserved for final validation at the end of the project.

Model

While deep learning is the focus of this paper, we have attempted to recreate the model used in the follow-up RPG's paper *2015 Post Level Term Lapse & Mortality Report*[2] to the best of our abilities and applied additional machine learning techniques to use as benchmarks.

As with any statistical learning model, we need to encode the categorical factors into numeric values (e.g., how do we represent "risk class is preferred nonsmoking"?). For our model, we apply a mix of one-hot encoding and embedding to transform the categorical variables. The structure of the neural net model includes a dense layer after the inputs, and then it splits off into two branches, each with another dense layer for the two outputs.

There is no "standard" on the network architecture, number of layers or number of neurons per layer. Modelers will often pick an initial model structure and run multiple iterations to arrive at an optimal model structure.

To quickly benchmark against traditional machine learning techniques, we apply automated machine learning (AutoML) to the data. AutoML fits multiple machine learning models (including random forests, gradient boosting machines [GBM], elastic net GLM and feedforward neural nets) with various hyperparameter combinations, within a user-specified time constraint, to determine the model with the best performance. For our case study, a GBM model is selected after five minutes of searching.

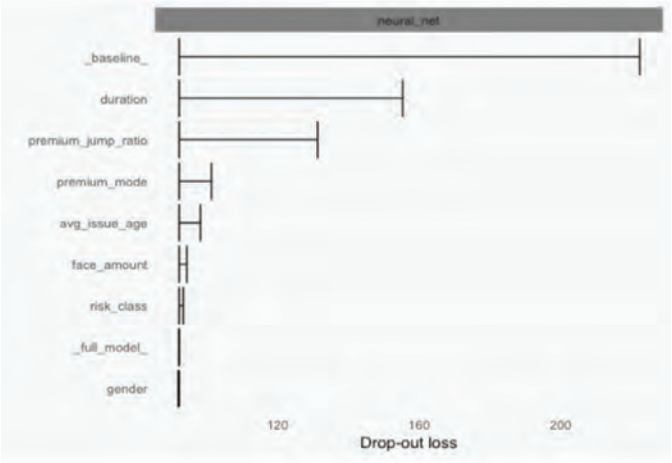
Performance

To measure and compare performance of the models, we use the weighted root mean square error (RMSE) metric applied to actual and predicted lapse rates. The weighted RMSE applies weights to errors of each block on the exposure of that block. The weighted RMSE for each of the models is as follows (lower is better): GLM (0.1722), AutoML (0.1619) and neural net (0.1695). The neural network and AutoML both perform better than GLM. In fact, AutoML performs the best with the least amount of work. We note that since this is an ongoing project, these metrics are calculated using the validation set. As more experiments are performed and we evaluate against the test set, we expect numbers to change. However, at this point, we see that the ML approaches are holding their own against a model built by industry experts.

PEEKING INTO THE BLACK BOX

While the machine learning and deep learning approaches outperform GLM in predictive accuracy, one common objection to implementing ML models in practice is that they are considered "black box" and impossible to explain. For some use cases, this doesn't matter. For instance, your favorite social media site is concerned more about whether you click on an ad and less about why you do it. The story is, of course, different in regulated industries such as insurance, where transparency is a core requirement.

Chart 1
Variable Importances for Neural Network Model



> Deep learning is a type of artificial intelligence that has been successfully applied in areas that involve large amounts of data.

Even for the same problem, the level of explainability requirements may change with the audience. As an example, for a pricing algorithm, your state regulator may have a higher bar than your underwriting team for transparency. In fact, sometimes they may have completely different definitions for what explainability is.

With the increasing adoption of ML methods in various fields, including "high stakes" applications in medicine and criminal justice, more and more research and software have focused on understanding the behavior of these black-box models.

In our case study, we experiment with a few (out of many possible) model explanation techniques. Some questions we try to answer are "What variables does the model think are important?" "How did the model come up with a particular prediction for the lapse rate?" and "What are the relationships between levels of a categorical predictor?" The plots we show are for the neural network model, although one can also construct them for the GBM and the GLM.

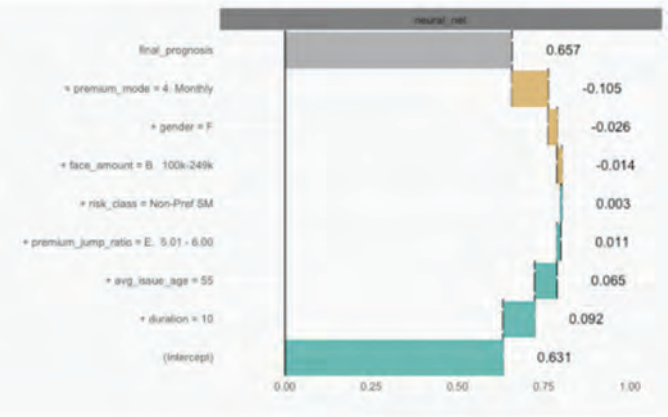
Variable Importance

In a linear model, one straightforward way to obtain predictor importance is to take the estimated coefficients and scale them by standard errors. In complex models such as neural networks, a common way to arrive at such a measure is to permute the values of the predictor of interest (thereby breaking the association with the response variable) and then see how much worse the model performs. In our neural network example, we see that duration and premium jump ratio turn out to be the most important variables, which is as expected. (See Chart 1)

Prediction Breakdown

There are also techniques to "break down" the prediction for a specific data point and approximate the contribution of each

Chart 2
Variable Attribution for a Single Lapse Rate Prediction

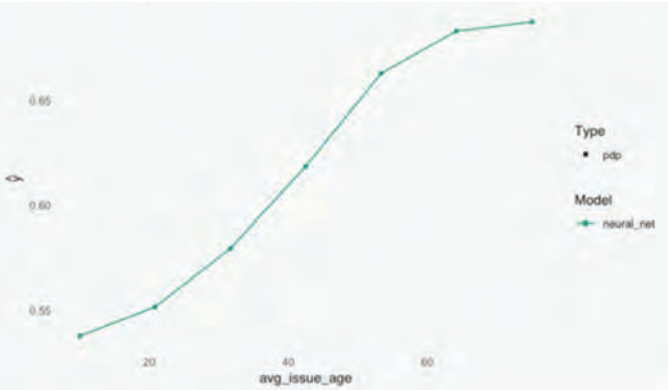


predictor to the predicted response. In this example, we can interpret from the plot that the average prediction for the data set is 63.1 percent lapse rate, but for this particular block, the prediction is higher due to the duration (immediately after the premium shock) and the issue age of 55, and these effects are partially offset by the monthly premium mode, arriving at a prediction of 65.7 percent. (See Chart 2)

Relationship Between a Predictor and the Response

We can construct a partial dependence plot (PDP), which tells us—all else being equal—how a change in one variable affects the response. From the PDP for issue age, we see that the predicted lapse rate tends to increase with issue age, with the effect tapering off at higher issue ages.

Chart 3
Partial Dependence Plot for Model Predicted Lapse Rate and Average Issue Age



All model interpretation techniques are wrong

It's important to keep in mind that model explanations, like the models they attempt to explain, are not exact. Each technique has its pros and cons. As an example, the PDP we show in Chart 3 can fall apart in the presence of highly correlated predictors. Even in the case of linear models like GLM, interpretation can be difficult if the predictors contain nonlinear transforms and interactions, as in the case of the SOA 2015 model.

POTENTIAL FOR DEEP LEARNING IN RELATED AREAS

A couple of insurance areas where deep learning may have some immediate applications are:

- **Data preparation:** This is an area where there has been limited success in automation; the fact that data provided to actuaries is already processed to some extent may make experience data amenable to automated cleansing.

  As just about any practitioner can attest, data preparation is typically the most onerous and time-consuming step in performing experience analysis. Data cleansing is definitely a nonlinear process that requires considerable judgment. Certainly the potential for more efficient and accurate data cleansing makes this a worthwhile area for future investigation.

- **Mortality deterioration:** Deep learning may also provide insight in modeling the extent of mortality deterioration, such as a Dukes-McDonald model.[2] Such a use would be easier to implement than data cleansing but would not provide as much value, since this approach would merely replace any processes that companies currently use for determining mortality deterioration. Nevertheless, deep learning may provide a way to automate these processes.

CONCLUSION

Based on the results we have developed using off-the-shelf deep learning technology, we believe that deep learning is a viable alternative to standard actuarial procedures for experience analysis. In particular, we note that the performance parameters indicate that results using deep learning compare favorably with standard techniques.

It's important to keep in mind that model explanations, like the models they attempt to explain, are not exact.

In addition, the effort required to implement our model was relatively mild, especially in the feature engineering and selection phase, which was mainly taken care of by the algorithms. In contrast, traditional model building using GLM requires multiple iterations by experts. ■

Kevin Kuo is a software engineer at RStudio and runs Kasa AI, kasa.ai, an initiative for open research in actuarial science. He is based in Seattle and can be reached at *kevinykuo@gmail.com.*

Bob Crompton, FSA, MAAA, is a vice president of Actuarial Resources Corporation of Georgia, located in Alpharetta, Ga.He can be reached at *bob.crompton@arcga.com.*

Frankie Logan is an associate with KPMG's risk consulting practice. He is based in Radnor, Penn. and can be reached at *flogan@kpmg.com.*

ENDNOTES

1 All the analysis is done using R, and the code is open source and available on the GitHub repository, ***https://github.com/kasaai/lapseml***. The neural network is built using the R interface to TensorFlow and Keras, ***https://tensorflow.rstudio.com/***. The AutoML model is implemented using H2O, ***https://cran.r-project.org/web/packages/h2o/index.html***. The model interpretability plots are created using DALEX, ***https://pbiecek.github.io/DALEX/***.

2 Lapse Modeling for the Post-Level Period—A Practical Application of Predictive Modeling. ***soa.org***, ***https://www.soa.org/research-reports/2015/lapse-2015-modeling-post-level/*** (accessed March 1, 2019).

3 1980. *Transactions of Society of Actuaries* 32, 547-584.