Retail Purchase Prediction with RFMAD

Chris Dulhanty
University of Guelph
cdulhant@uoguelph.ca

Alexander Hinton
University of Guelph
ahinton@uoguelph.ca

Abstract

This paper expands upon the commonly used recency, frequency, monetary (RFM) model of customer behaviour in a retail purchase prediction problem in order to demonstrate improved performance via the introduction of machine learning methods. Five simply constructed customer features - recency, frequency, monetary, amount and duration (RFMAD) - are used as inputs into three machine learning models with the goal of forecasting next period purchases. A logistic regression, deep neural network (DNN), and recurrent neural network with long-short term memory cells (LSTM) are applied to a dataset from a large healthy and beauty retailer, containing transactional information of one million customers over a two year period. Performance is measured via the mean of the area under the receiver operating characteristic curve (AUROC). The LSTM model is found to be the best performer with 0.7563 AUROC, followed closely by the DNN and logistic regression, with 0.7528 and 0.7522 AUROC, respectively. All three models demonstrate notable improvement over the benchmark RFM model.

1 Introduction

Retail purchase prediction is a problem that impacts all retailers of goods or services. The ability to forecast the buying habits of customers allows for many strategic advantages for a company. At an individual customer level, marketing efforts can be streamlined towards targeting those individuals who are predicted as likely to purchase certain goods or services, and at a macro level the individual predictions can be amalgamated to help optimize inventory and staffing decisions based on expected demand. Financial forecasts can be made based on enhanced revenue projections and can enable more accurate allocation of funds.

Machine learning has proven to be a very powerful tool for pattern extraction and prediction in a variety of domains. Both linear models and nonlinear models have been developed to address the varying complexity of real-world phenomena. In addition, success has been seen in deploying these models on both structured and unstructured data. In the case of retail purchase prediction, data is mainly in the former state, as transactional, customer and product information is logged in relational databases.

Current methods in retail purchase prediction are predicated on the collection of data from a customer through a unique customer identification. Loyalty programs are very common among retailers to obtain the longitudinal data of a customer's purchasing habits. One of the most common models of customer engagement is called the recency, frequency, monetary (RFM) model. This model scores a customer in these three areas by aggregating their purchasing habits and bins them into groups from one to five on each of the three characteristics [1, p.2,3] . Predictions are then made by summing an individual customer's scores. This method of prediction is very common in the field due to its effectiveness and to its ease of implementation [2, p.1], however it does not have any inherent learning aspect to it and thus is ripe for improvement with machine learning techniques.

The aim of this paper is to leverage the existing RFM data that is commonly tracked by retailers to develop a new retail purchase prediction model. Through the addition of two new features and the application of more powerful machine learning techniques to the data, the goal is to create a retail prediction model that can be implemented with limited overhead in further data processing. In addition, as this model will only look at the transactional information of a customer, it is product agnostic. This will hopefully allow for easy transfer learning to other areas of the retail sector.

This approach to retail purchase prediction will be used to predict the purchasing habits of one million customers from a large health and beauty retailer. Specifically, given two years of transactional data, the goal is to predict purchases for each of the five top-brands in the 31-day period following the final date provided in the dataset. Predictions will be measured by the mean of the area under the receiver operating characteristic curve (AUROC) for each of the five brands on an unseen test dataset containing the actual transactions of these customers in the next 31 days. A benchmark RFM model is compared to a logistic regression, a deep neural network (DNN) and a long-short term memory (LSTM) network. To the author's knowledge this is the first paper employing LSTM networks on retail purchase prediction in the health and beauty space.

The remainder of this paper is organized in the following manner. Section 2 highlights relevant work in the field of retail purchase prediction and machine learning and provides context for this paper. Section 3 provides information on the dataset used in the experiment. Section 4 outlines model architectures and the results of their application to the problem at hand. Section 5 provides discussion of results followed by a brief conclusion in Section 6.

2 Relevant Work

Baesens et al. studied purchase modelling on a dataset of 100,000 customers from a major European mail-order retailer. Using RFM characteristics of customers, they compared the performance of linear discriminant analysis, quadratic discriminant analysis, logistic regression, and Bayesian neural networks, with the goal task of predicting whether a customer would or would not purchase a product in the next time period [3, p.200] . The model's performances were compared on percent accuracy, as well as AUROC. The authors found Bayesian neural networks were the top performing models, followed by logistic regression [3, p.204]. They extended their experiments by augmenting the feature set with non-RFM variables (such as the length of the customer/retailer relationship among others) and demonstrated improved modelling performance when these non-RFM variables were included [3, p.206]. These results led us to include logistic regression as well as a neural network as models in our experiments, as well as to augment the feature set with non-RFM variables.

Kootie et al. studied online consumer behaviour, as recorded by email purchase receipts. From a dataset containing 20.1 million users and their 121 million purchases from February to September 2014, they trained a Bayesian Network Classifier to predict the timing and price of the customer's next purchase from a set of possibilities, modelling the problem as classification [4, p.2,7]. Notably, they found that user demographics were not particularly useful in predicting a customer's next purchase, but that temporal factors were the most important features [4, p.8]. This lead us to believe that using a model tailored to the processing of temporal data would be very useful in predicting future purchases.

Recurrent neural networks (RNNs) are a type of neural network that specializes in processing sequential data. Input to the hidden layer at time t is a function of the input from the below layer at time t and the input from the hidden layer at time $t-1$. These recurrent pathways allow RNNs to share parameters through a deep computational graph [5, p.372-376]. One challenge that RNNs face, however, is that gradients either vanish or explode over time as errors are propagated through the length of the network. Even if gradients are in an acceptable range, long-term dependencies are difficult to establish as the magnitude of weights are very small. The long short-term memory model of the RNN has been proposed as a solution to this issue. LSTM cells employ self-loops to create pathways where the gradient can flow for long durations and have been proven successful in many applications with temporal components, including speed recognition and image caption [5, p.404-407].

The application of LSTM models to consumer behaviour prediction was investigated by Salehinejad et al. in their study of the Ta-Feng Grocery Store dataset, containing 32,266 customers, 817,741 transactions and 23,812 products [6, p.5]. They calculated recency, frequency and monetary values for customers on weekly intervals, and used these values, along with the customer loyalty number, as input to vanilla RNNs and LSTM models with various activation functions. RFM values of customers in the next time period could be best predicted by LSTM models with rectified linear (ReLU) activation functions [6, p.6]. This paper aims to extend this work from predictions of RFM values to the prediction of actual customer purchases.

3 Data

3.1 Data Source

A proprietary dataset of transactions from a large health and beauty retailer was provided by Rubikloud Technologies Inc., a venture-backed startup that helps retailers leverage machine learning and big data. The dataset was comprised of transactional information for a sample of one million customers in the company's loyalty program for purchases from January 1st, 2015 to December 31st, 2016. Transactional information contained 18,198,302 sales on a product-level, containing date, quantity, price and contents of each transaction. Product information was provided with price and brand information, along with four levels of categorization. Customer information was provided with the date of registration in the retailer's loyalty program. Five brands were specified as the top brands at the retailer, for which predictions would be made. These brands represented 4,615,262 sales and 25.3% of all transactions for the one million customers provided.

3.2 Train and Test Splits

A training set was carefully created from the provided dataset to adhere to the seasonality of the prediction problem. Features were extracted from transaction data in the time-period January 1st, 2015 to December 31st, 2015 and labels were extracted from data in the time-period January 1st, 2016 to January 31st, 2016. Transaction data was aggregated on a customer-level so each customer represented one example to make a prediction on.

In order to ensure an appropriate train-test match, customers in the dataset were evaluated by their date of registration. Customers registered after December 31st, 2015 were excluded from the training set, as these customers were not in the system in calendar year 2015 and therefore had no features to make a prediction on. This resulted in a total of 617,742 customers to be evaluated in the training process. It should be noted that in the provided benchmark model from Rubikloud Technologies Inc. this customer validation operation was not performed. The benchmark model made predictions on all one million customers based on features from calendar year 2015, although more than a third of the customers did not exist in their database at the time. We adjusted the benchmark model and have reported its performance in Section 4.

Features for the test set were extracted from the time period January 1st, 2015 to December 31st, 2016. Labels for the test set were to be extracted from the 31-day period following the final day in the dataset, January 1st to January 31st, 2017, however this data was not provided by Rubikloud Technologies Inc.. We were therefore not able to report the final performance of our models on the problem as outlined. References to the performance of our models are based on a 5-fold cross validation on the training set.

3.3 Feature Extraction

Aggregations were performed on a customer-level to define features. Four features were extracted from the transaction data for each brand:

- Recency: number of days from the final day in the time-period to the last purchase of the customer
- Frequency: count of unique transactions a customer made in the time-period
- Monetary: sum of products purchased in the time-period
- Amount: count of total products a customer purchased in the time-period

A fifth feature was extracted as the number of days since registration, which we call Duration. Each customer therefore contained four features for each of the five brands and one additional feature for a total of 21. We refer to these features as RFMAD.

Features were normalized using Scikit-learn's MinMaxScaler preprocessing operation to be within the range of 0 to 1.

4 Experiments

4.1 Benchmark RFM Model

Our benchmark model is the RFM model provided by Rubikloud Technologies Inc.. This model ranks customers on each of their RFM characteristics for each brand, and allocates them into decile bins. Customer predictions are made by summing a customer's bin number for each feature and dividing by 30. Hence if a customer is in bin 10 for R, F and M, they would have a probability estimate of 1.0 for the next purchasing period. Given that the model is fixed and there is no learning occurring, we do not need to do cross validation, and can make predictions on the full training set simultaneously. The results of this model are shown in column two of Table 1. The mean AUROC across all five brands for the benchmark model is 0.7240.

4.2 Logistic Regression

A logistic regression classifier was designed for multi-label output using Scikit-learn's logistic regression and One-vs-rest classifier functions. Each 21-dimensional example was fed into the model with the goal of predicting the five brand-level binary labels. The model employed the logistic loss function and the 2-norm weight penalty, and regularization strength was chosen via grid search. In the Scikit-learn logistic regression model, the hyperparameter C is a constant equal to the inverse of regularization strength, such that small values of C specify stronger regularization. This loss function is given in Equation 1.

$$Loss = \min_{w \in R^{\mathrm{d}}} \sum_{i=1}^{N} log(1 + e^{-y_{\mathrm{i}} f(\mathbf{x}_{\mathrm{i}})}) + \frac{1}{C} \|\mathbf{w}\|^2 \tag{1}$$

The value of C was selected via grid-search over the domain:

$$\text{C=}10^i,\, i \in [-5, -4, -3, -2, -1, 0, 1, 2, 3, 4].$$

At each setting of C, five-fold cross validation was performed, where the model was trained on 80% of the training set, and tested on the remaining 20% of observations (the validation set), with the validation set changing in each fold. Due to the imbalanced nature of the data, class weighting was incorporated into the model. The overrepresented class (examples with negative labels) was given an importance weight of 1, while the underrepresented class (examples with positive labels) was given an importance weight of 1 / (proportion of purchases in the training set), which worked out to be an importance weight of approximately 37.

Mean training and mean validation AUROC were calculated at each setting of C, and the graph of train/validation AUROC across decreasing regularization strength (increasing C) is shown in Figure 1.

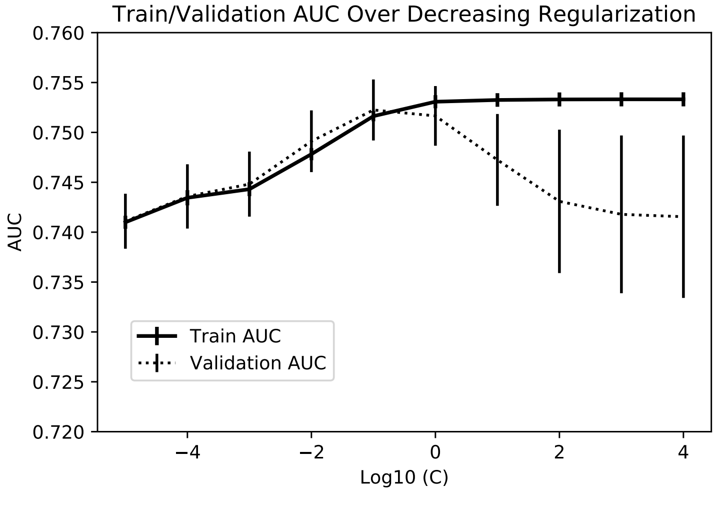

Figure 1: Mean ($\pm\sigma$) training AUROC and mean ($\pm\sigma$) validation AUROC from five fold cross validation plotted against $\log_{10}$ (C) from the logistic regression model. C is the inverse of regularization strength, therefore as C increases, the regularization of the model is decreasing.

Initially as C increases (regularization decreases), both training and validation AUROC are increasing. However, as C becomes larger than 10^{-1} overfitting occurs: mean training AUROC continues to rise, while mean validation AUROC declines sharply. Therefore $C = 10^{-1}$ was selected as the optimal setting for the inverse of regularization strength hyperparameter. The results of this model are displayed in column 3 of Table 1. The mean AUROC of the optimized logistic regression model was 0.7522, with a standard deviation of 0.0031 from the 5-fold cross validation. Assuming that the distribution of scores is approximately Gaussian, a 95% confidence interval for mean AUROC for the logistic regression model is [0.7461, 0.7583].

4.3 Deep Neural Network

A multi-layer feed-forward artificial neural network was designed for multi-label output using Keras with a TensorFlow backend. 21 features were extracted from the time-period January 1st to December 31st, 2015. Each 21-dimensional example was fed forward through a number of hidden layers with ReLU activation function to a five-unit output layer with a sigmoid activation function. 50% dropout was employed between hidden layers and between the hidden layer and output layer for regularization. Binary cross entropy was used as the loss function.

To optimize hyperparameters, random search was first employed, as it has been shown by Bergstra et al. to find better models than grid search or manual search by effectively searching a larger configuration space, given the same computational budget [7, p.302]. A 20-iteration coarse random search was run, employing 5-fold cross validation in every model to determine performance. The learning rate was drawn from a logarithmic distribution between 0.1 and 0.0001, the number of layers was drawn from a uniform distribution between 1 and 3 and number of units in each hidden layer were drawn from logarithmic distributions between 1 and 256. Mini-batch training was employed with a batch size of 1024. The Adam optimization algorithm was utilized. Each model was trained for 25 epochs.

After random search was complete, further hyperparameter optimization was performed with a finer search criteria, based on the above results. The SigOpt API was used, employing ensemble Bayseian Optimization techniques for 20 iterations. The final model was found to be have two hidden layers, a learning rate of 0.00241, and 230 hidden units in the first layer and 96 in the second.

Mean training and mean validation AUROC were calculated after each epoch, and the graph of train/validation AUROC across epochs is shown in Figure 2.

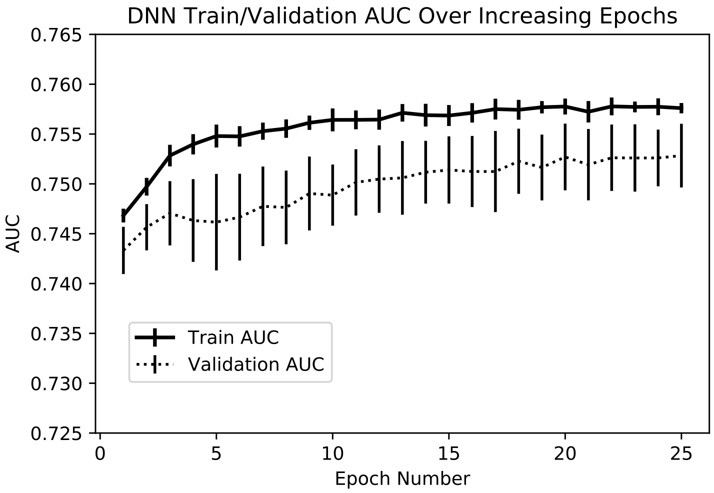

Figure 2: Mean ($\pm\sigma$) training AUROC and mean ($\pm\sigma$) validation AUROC from five fold cross validation plotted against epoch number for the deep neural network model.

Both training and validation AUROC climb steadily for the first 20 epochs before levelling out. Validation AUROC does not decrease in this epoch range so overfitting has not occurred. The results of this model are displayed in column 4 of Table 1. The mean AUROC of the optimized deep neural network was 0.7528, with a standard deviation of 0.0032 from the 5-fold cross validation. Assuming that the distribution of scores is approximately Gaussian, a 95% confidence interval for mean AUROC for the deep neural network is [0.7465, 0.7591].

4.4 LSTM network

A multi-layer recurrent neural network with LSTM cells was designed for multi-label output using Keras with a TensorFlow backend. 21 features were extracted for each calendar month in 2015. Each example was fed into the network as:

$$\mathbf{x} = [\mathbf{x_{jan}}, \mathbf{x_{feb}}, \mathbf{x_{mar}}, \mathbf{x_{apr}}, \mathbf{x_{may}}, \mathbf{x_{jun}}, \mathbf{x_{jul}}, \mathbf{x_{aug}}, \mathbf{x_{sep}}, \mathbf{x_{oct}}, \mathbf{x_{nov}}, \mathbf{x_{dec}}]$$

Where each $\mathbf{x_i}$ is the 21-dimensional features for that month. The target label was the 5-brand purchase predictions for January 2016. In this setup, the model can learn the sequences of monthly RFM characteristics which best predict transactions for January 2016. 50% dropout was employed between hidden layers and the hidden layer and output layer for regularization. Binary crossentropy was used as the loss function.

Once again, a 20-iteration coarse random search was used to optimize hyperparamters, followed by 20-iteration finer Basesian Optimization using the SigOpt API. In both cases, 5-fold cross validation was used in every model to determine performance. In random search, the learning rate was drawn from a logarithmic distribution between 0.1 and 0.0001, the number of layers was drawn from a uniform distribution between 1 and 2 and number of units in each hidden layer were drawn from logarithmic distributions between 1 and 128. Mini-batch training was employed with a batch size of 1024. The Adam optimization algorithm was utilized.

The final model was found to be have two hidden layers, a learning rate of 0.00054, and 76 hidden units in the first layer and 75 in the second.

Mean training and mean validation AUROC were calculated after each epoch, and the graph of train/validation AUROC across epochs is shown in Figure 3.

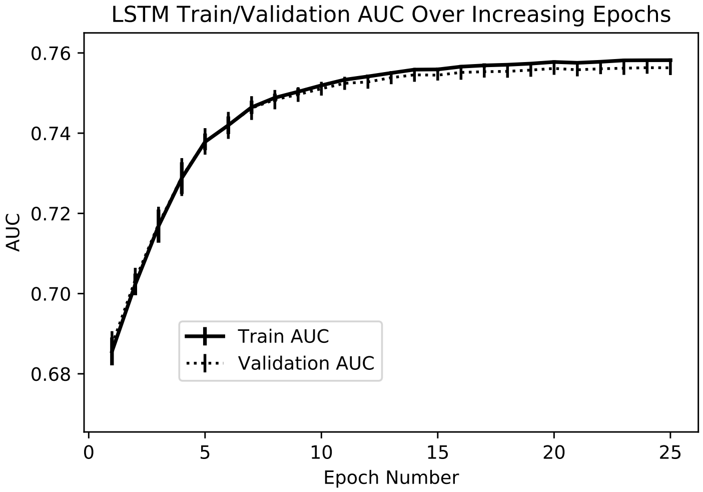

Figure 3: Mean ($\pm\sigma$) training AUROC and mean ($\pm\sigma$) validation AUROC from five fold cross validation plotted against epoch number for the LSTM network model.

The results of this model are included in column 5 of Table 1. The mean AUROC of the optimized LSTM network was 0.7563, with a standard deviation of 0.0018 from the 5-fold cross validation. Assuming that the distribution of scores is approximately Gaussian, a 95% confidence interval for mean AUROC for the LSTM network is [0.7528, 0.7598].

Table 1: Model Performance Comparison via AUROC

Model AUROC				
Brand	Benchmark RFM	LR	DNN	LSTM
11	0.6726	0.6997 ± 0.0063	0.7034 ± 0.0058	0.7062 ± 0.0052
18	0.7040	0.7446 ± 0.0052	0.7440 ± 0.0046	0.7455 ± 0.0047
46	0.7162	0.7509 ± 0.0065	0.7522 ± 0.0012	0.7559 ± 0.0066
52	0.7711	0.7863 ± 0.0071	0.7839 ± 0.0071	0.7917 ± 0.0033
190	0.7560	0.7798 ± 0.0023	0.7807 ± 0.0068	0.7822 ± 0.0017
MEAN	**0.7240**	**0.7522 ± 0.0031**	**0.7528 ± 0.0032**	**0.7563 ± 0.0018**

5 Discussion

For all three optimized machine learning classifiers, the lower limit of the 95% confidence interval for mean AUROC is greater than the AUROC achieved by the benchmark RFM model. Therefore we can conclude with 95% confidence that the machine learning models show significant performance improvement relative to the benchmark, which was the main aim of the project. The average AUROC curve is shown in Figure 4.

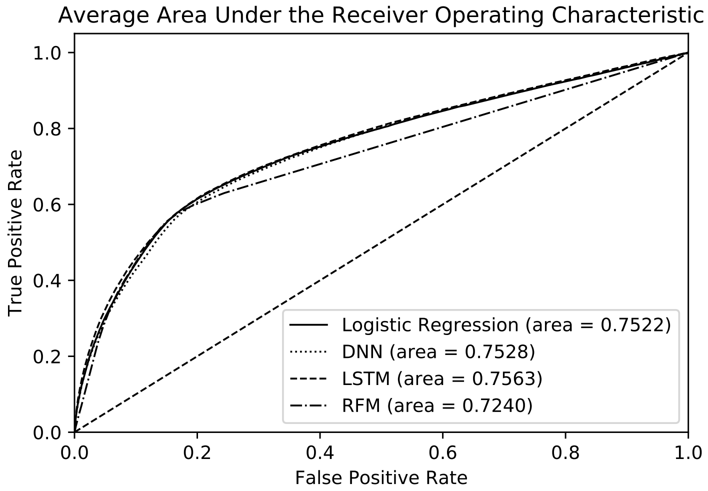

Figure 4: Average AUROC curve for the four classifiers.

Determining which of the three models is best is a trickier task. As seen in Figure 4, the AUROC curves are almost indistinguishable. While the LSTM network had the highest mean AUROC of 0.7563, the confidence intervals for all three models are overlapping. To determine if significant performance differences can be found between classifiers, t-tests were done for all three possible comparisons of ML classifiers. The t-statistics were calculated as follows:

$$t = \frac{(\mu_{\mathrm{x}} - \mu_{\mathrm{y}})}{\sqrt{\frac{{s_{\mathrm{x}}}^2}{n} + \frac{{s_{\mathrm{y}}}^2}{n}}} \tag{2}$$

Where μ_{i} and s_{i} are the mean and standard deviation for model i in the 5-fold cross validation, and n=5. The results of the three t-tests are shown in Table 2.

The LSTM classifier demonstrates significantly greater performance at the 10% significance level than the Logistic regression model and DNN, however neither of the t-statistics are significant at the 5% level (t > 2 is usually considered the threshold for significance at the 5% level, however due to only 4 degrees of freedom the critical value of t at the 5% significance level is 2.13). It appears the LSTM model is the best classifier of the three, however the evidence is not striking.

One important note is that the benchmark model included only the three RFM features, while the machine learning models included five. Therefore, the difference in performance could be due to

Table 2: T-tests between classifiers

Comparison	T-Stat
LSTM- Logistic	2.09*
LSTM- DNN	1.75*
DNN - Logistic	0.24

*significant at 10%, **significant at 5%, ***significant at 1%. P-values calculated based on 4 df.

improved features, or improved models. A robustness check (not displayed) was performed with the same methodology of the benchmark RFM model, but with all five RFMAD features. This model gave a mean AUROC of 0.7234, slightly worse than the RFM benchmark. This demonstrates that the improved performance is not simply due to the improved features, but was due to the pattern extracting and predictive abilities of the machine learning models.

Although the evidence is not striking, it appears the LSTM model is the best classifier of the three. This is not surprising given the temporal nature of the dataset and the proven success of LSTM models in many applications with sequential components.

6 Conclusion

In this paper we focused on one-month-ahead retail prediction from a major retailer in the health and beauty space. Three machine learning prediction models (logistic regression, deep neural network, recurrent neural network) were constructed with industry standard RFM features, as well as two additional features which we call Amount and Duration. These features were calculated for each of the five top brands of the retailer over the in-sample training period January 1st to December 31st, 2015, with binary labels determined from transactional data in January 2016. The performance of the three machine learning models was compared to a benchmark model provided by Rubikloud Technologies Inc., which put customers into decile bins based on their scores in each category, and made predictions based on bin summation.

The three machine learning models had similar performance, with mean AUROC between 0.7522 and 0.7563, all significantly greater than the mean AUROC of 0.7240 generated from the benchmark model. Statistical tests were performed to compare performance between all pairs of models, and it was concluded at the 10% significant level that the LSTM model outperformed the logistic regression and the deep neural network. Retailers who are already tracking customer segmentation variables such as RFM can make demonstrable improvements in their predictive power by including machine learning models in their retail purchase forecasts, and for best results, it is recommended an LSTM model is used such that the temporal nature of the data can be harnessed.

References

[1] Miglautsch, J. R. (2000). Thoughts on RFM scoring. *Journal of Database Marketing & Customer Strategy Management*, 8(1), 67-72.

[2] Yeh, I. C., Yang, K. J., & Ting, T. M. (2009). Knowledge discovery on RFM model using Bernoulli sequence. *Expert Systems with Applications*, 36(3), 5866-5871.

[3] Baesens, B., Viaene, S., Van den Poel, D., Vanthienen, J., & Dedene, G. (2002). Bayesian neural network learning for repeat purchase modelling in direct marketing. *European Journal of Operational Research*, 138(1), 191-211.

[4] Kooti, F., Lerman, K., Aiello, L. M., Grbovic, M., Djuric, N., & Radosavljevic, V. (2015). Portrait of an Online Shopper: Understanding and Predicting Consumer Behavior. *arXiv preprint* arXiv:1512.04912.

[5] Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT press.

[6] Salehinejad, H., & Rahnamayan, S. (2016, December). Customer shopping pattern prediction: A recurrent neural network approach. *In Computational Intelligence (SSCI), 2016 IEEE Symposium Series on* (pp. 1-6). IEEE.

[7] Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13(Feb), 281-305.