Forecasting Family Consumer Decisions in the United States

Jared Azevedo & Andrés Suárez
School of Engineering & Law School
Stanford University
jaredssm@stanford.edu & asuarezg@stanford.edu

1. Introduction

Have you ever wondered how a company decides what product to release next? Historically, a combination of market research tactics might be used to see what consumers were interested in buying next. Today, the power of machine learning has opened the door to using historical consumer data to more accurately predict what someone wants or needs to buy next. It has also helped to visualize trends in the marketplace which is another important factor companies might consider. While this works well for entire marketplaces, it becomes a bit more nuanced when we try to look at particular market segments and goods.

That is where we step in - to make a model that can perform well at predicting family consumer decisions on specific goods. The family unit is a much smaller piece of market segments and allows us to better predict for a larger variety of individuals. A wide swath of demographics will be included as well as a decent variety of goods, resulting in a much more zoomed-in scale than other consumer habit machine learning projects. We will build off of existing efforts in this space and give companies a tool that can be used to answer questions about specific family backgrounds instead of wider market segments! This is a powerful application that could influence how goods are branded and marketed, or which goods even come to see the light of day (i.e. get produced). For example, this could help reveal what factors influence a family's decision to purchase a good and to what degree that factor is important relative to other influences. The input to our model is text data consisting of demographics and purchasing history which gives an output saying whether that individual will purchase a specific good or not.

2. Related Work

As part of our research, and despite conducting a thorough review of existing literature, we found few studies that aim to predict individuals' or families' retail goods demand decisions through the use of machine learning. On the other hand, we did find more studies that looked to predict demand decisions in other markets, especially in the energy and water markets. From our understanding, the absence of a more significant number of studies focused on retail goods' demand could be explained by the difficulties of finding publicly accessible data characterizing individuals' or families' consumption decisions with a meaningful number of features and observations to allow the researcher take advantage of the capabilities of machine learning methodologies, especially deep learning. The studies that aimed to predict consumers' retail goods purchase decisions are Toth et al. (2017), Kiran et al. (2021), Ibrahim (2022), Saha et al. (2022), and Punia et al. (2020). On the other hand, the study that conducted a similar task but for water consumption was developed by Kim et al. (2022).

Toth et al. (2017) analyzed live shopping sessions to predict three outcomes: purchase, abandoned shopping cart, and browsing-only. From the perspective of the authors, despite the increasing importance of online shopping, it is still important to understand the reasons explaining the lower conversion rates (share of likely consumers that complete a purchase) for this type of shopping compared to more traditional methods. The authors used high-order Markov chains and recurrent neural networks for this task and found that for sequences truncated to 75% of their length, a relatively small feature set predicts purchase with an F-measure of 0.80. Kiran et al. (2021) use machine learning to predict consumer behavior on social media platforms such as Linkedin, Facebook, Instagram, and Youtube, in terms of likes, followers, visits, and downloads. The researchers use linear regression, decision trees, random forest, and extra tree regression, amongst other methods, to evaluate, for example, the relationship between the previous variables of a product on a platform, for example, YouTube, with the behavior of the same variables for the same product or brand another platform, such as Facebook. Finally, the authors concluded found that the best model to perform this task was decision trees with an accuracy on the test data of 98%.

Ibrahim (2022) aimed to predict the moment at which consumers purchase retail goods using methodologies such as decision trees, random forest classifiers, and support vector machines. Predictions about the times when consumers were expected to purchase retail goods were planned to be used to offer promotions to incentivize the purchase of specific goods. The authors found that the method with the highest performance was the random forest classifier. Saha et al. (2022) looked to predict future seasons of retail goods' demand using long-short term memory and light gradient boosting. According to the authors, improving companies' ability to predict demand is becoming more important as a result of increasing competition, the shortage of employees, and cost optimization, among other reasons. The authors used historical data sales from an American multinational retail company. The authors found that light gradient boosting had a better performance than the long-short-term memory model. Finally, Punia et al. (2020) aimed to forecast retail goods' demand considering how important is for companies to make more informed decisions in purchasing, inventory management, scheduling, capacity management, etc. The authors evaluated their method, a combination of long short-term memory networks and random forest, against other more popular and widely used methodologies using multi-channel retailer data. This data included among its features information about sales, products, and stores. The authors concluded that the method they proposed outperformed more traditional ones such as random forests, neural networks, multiple linear regression, and LSTMs.

Finally, Kim et al. (2022) studied water consumption at the household level in the United States. According to the authors, the pattern of water consumption can vary based on phenomena such as weather and holidays and this variation has limited the predictive ability of models such as autoregressive integrated moving averages. The authors used both long-short-term memory approach (LSTM) and autoregressive integrated moving average (ARIMA) to predict water consumption at the customer level. The authors found that the LSTM model performed better than the ARIMA for all four different water-use types considered in the study (detached houses, apartments, restaurants, and elementary schools).

To conclude, in this literature review we characterized existing research in the field of retail goods' demand forecasting. Despite the existence of some studies, from our perspective, we could not find any study following exactly our approach, that is, using families' demographics to predict their consumption patterns. Consequently, we could not use directly their methodologies or findings to substantiate our research. Additionally, we believe the absence of studies similar to ours could be explained by the difficulty of finding big databases including both consumption information and demographic variables. Also, it could be possible that companies are already using this type of methodology using approaches similar to ours without information about these applications being available to the public.

3. Dataset and Features

Due to some technical and privacy limitations, we had to switch what dataset we were using from the proposal. Instead, we are using a dataset from the U.S. Bureau of Labor Statistics called the Consumer Expenditure Survey. The data is gathered via a series of surveys distributed to a selection of families that rotates every couple of weeks (it varies between the different types of surveys they conduct). The Consumer Expenditure Survey program has been going on for a couple of decades, but we are opting to use data from 2008 to 2021 only. Furthermore, we decided to use the Diary survey responses instead of the Interview survey responses because the Diary survey responses included a better breakdown of information that we want to use and in a more accessible format (albeit a lot of preprocessing is still required).

Within each year, the results are stored across several different kinds of files and each kind of file shows up four times corresponding to each quarter of that year. For example, one kind of file (FMLD) stores income and demographic information about the respondents while another kind of file (EXPD) stores their responses to what purchases they have made recently. The data itself is primarily stored as numerical values which are then mapped to real-life meaning in a separate, overview file (like a product code to what that product actually is) but have also been appended with letter columns that indicate metadata about the responses. We have pruned the letter columns in order to only select the numerical data we are interested in using for our model. We have also merged the different types of files together in order to represent each survey response holistically (store all demographic and purchasing information together in one example). Finally, we have backfilled the data so that there is an accurate portrayal of both purchases made and not made (only purchases made are tracked on the survey).

Despite earlier suggestions to not mix multiple years' worth of data for our training data, we did ultimately mix multiple years and then split the mixed data into our training/validation/test sets. Once we compiled data from more than one year, we split the data such that 90% was training data, 5% was validation data, and 5% was test data. We could have gone with an even larger share of training data, but we decided not to so that we did not overfit our training data (plus the total amount of data we had was relatively sparse since we had to do a lot of backfilling of missing values). That said, the total number of examples (before splitting into subsets) was 160,169 with 42,447 positive labels and 117,722 negative labels. Note that the labels vary depending on which good we are running the model for (these numbers are for cookie products). Below is an example of what examples looked like to the model (i.e. after all of the preprocessing):

PURCHASED	NEWID	UCC	AGE_REF	AGE2	ALCBEV	BAKEPROD	BEEF	...	WTREP03	WTREP04	WTREP05	WTREP06	WTREP07	WTREP08	WTREP09	WTREP10
0.0	4882432.0	20510.0	18.0	0.0	88.0	2.0	0.0	...	279111.0	0.0	0.0	0.0	380102.0	363400.0	388910.0	309875.0
0.0	4718602.0	20510.0	23.0	0.0	0.0	2.0	0.0	...	0.0	93815.0	0.0	0.0	0.0	121029.0	105024.0	111117.0
0.0	4882461.0	20510.0	38.0	0.0	0.0	2.0	0.0	...	0.0	132079.0	125707.0	109328.0	0.0	0.0	148135.0	0.0
0.0	4882462.0	20510.0	38.0	0.0	0.0	0.0	41.0	...	0.0	132079.0	125707.0	109328.0	0.0	0.0	148135.0	0.0
0.0	4784161.0	20510.0	33.0	35.0	41.0	8.0	8.0	...	88316.0	69930.0	77653.0	0.0	95207.0	0.0	0.0	0.0

In all, we have 73 features along with the label and NEWID.

4. Methods

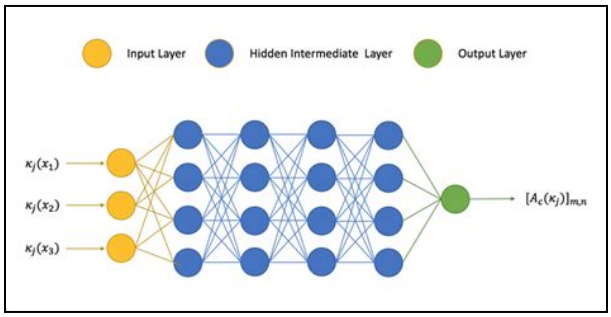
Based on the structure of our dataset, we decided to use a conventional deep neural network. As previously mentioned, our approach is novel in the sense that we could not find any other studies trying to predict retail goods' demand using consumers' demographic variables. Other studies have tried to predict demand using databases with a time component, that allows the use of other machine learning methods such as LSTMs. However, the Consumer Expenditure Survey data only followed each family for a two week period; time frame that we did not consider long enough to allow us study family consumption

decisions over time. Additionally, the structure of our data limited our ability to use other methods that are better suited for analyzing other types of data, for example images, sound or text.

The deep neural network we implemented had the following structure:

- An input layer with size equal to the number of different demographic variables used to train our model.
- A first dense layer with 32 neurons and relu activation function.
- A dropout layer with dropout rate equal to 25%.
- We repeated the two previous layers for four consecutive times and added a final layer with only one neuron and that uses the sigmoid function as its activation.

Deep neural network in contraposition to shallower methods, such as logistic regression, increase the function's ability to more complex data structures. The following is the the representation of the neural network similar to ours but without using a dropout layer after every dense layer.



Source: [An-illustration-of-a-deep-neural-network.ppm (850×432) (researchgate.net)]

In terms of the loss function, we took two different approaches. First, we used for our baseline model the binary cross-entropy loss. Second, we created additional loss functions to evaluate if they could improve the performance of our model measure through the five metrics used to evaluate the model. The following are the loss functions we created:

1. $Loss = \frac{Y_{True}}{Y_{Pred}+\xi} + \frac{1-Y_{True}}{1-Y_{Pred}+\xi}$
2. $Loss = -1/m \sum_{i=1}^{m}((1 + y_{i,true}) \bullet log(2 + y_{i,pred})) + (y_{i,true} \bullet log(1 + y_{i,pred}))$
3. $Loss = y_{true} \bullet y_{pred}$
4. $Loss = 1/m \bullet (y_{true} - y_{pred} + 0.5)^2$
5. $Loss = 1/m \bullet (y_{true} - y_{pred} - 0.5)^2$

Our goal when experimenting with these different loss functions was to increase the performance of our model, which did occur! Functions 1, 2 and 4 increased resulted in great recall scores at the expense of precision. On the other hand, functions 3 and 5 resulted in good accuracy, but this came at the expense of every other metric performing worse or equal to the baseline model. The results for all the different loss function are available in the appendix.

The idea behind all of these loss functions but the first one was to increase the weighting in some direction. For function 1, we created a function that would return a high value in case of mismatches

between the true and predicted values and a value of 1 when the prediction was accurate. For function 2, we wanted to increase the weight towards true labels and add noise even when the label was zero. For function 3, we wanted to only factor true labels into the equation and dismiss any predicted or true zero labels. For functions 4 and 5, we wanted to add either a positive or negative weight to the standard mean squared error.

5. Experiments/Results/Discussion

To start, let's discuss the hyperparameters we experimented with and ultimately settled on for our model. For the learning rate, we chose 0.00001. This is a lower-than-standard learning rate, but we opted to go for it because we observed that our model would quickly diverge otherwise. For our batch size, we went with 128 because we found that a smaller, but not too small value produced quick training without compromising the integrity (performance) of our model. For our epochs, we went with 20 as it did not take many epochs for the training performance to converge and stabilize. Ultimately, we discovered that choosing different combinations of hyperparameters didn't really alter the performance of our models but mostly affected their training speed.

Next, let's talk about the metrics we decided to focus on. For this problem, we extracted the accuracy, recall, precision, and AUC. During some experiments, we also looked at F1 score but since the performance of the model was quite poor it did not help us evaluate any more than the primary metrics we already chose. On our best model configuration, we got the following metrics:

```
Test loss: 54.828128814697266
Test accuracy: 52.515918016433716
Test AUROC: 0.5017833113670349
Test precision: 26.83923840522766
Test recall: 9.394372999668121
```

If anything, we probably underfit our training data. We attempted to increase the amount of data and the split that goes to training data, but to little avail. Below is the confusion matrix we generated on our test data:

```
Confusion matrix:

tf.Tensor(
[[5505  407]
 [1960  137]], shape=(2, 2), dtype=int32)
```

The above results were obtained when we used a mix of demographic and socioeconomic features in our data. Prior, we used just a small collection of demographic features and observed even worse performance. We tried different combinations of features and found that once we reached 73 features there was no notable change in the performance of the model. However, further feature engineering could reveal that some do indeed improve performance!

6. Conclusion/Future Work

At the end of the quarter, we found that the deep neural network with all the features available is the highest-performing. We believe this version of the algorithm had the best performance because we were providing more information about each family to the model to make the prediction about their consumption decisions. In future work, we would love to add more team members and explore in greater depth what can be done with the dataset that we initially had planned to use and which includes more features, observations and follows consumers over longer periods of time (therefore, allowing the study of more meaningful questions and the use of more advanced models).

7. Contributions

Jared helped write the majority of the data preprocessing code, created and wrote most of this final report save section 2 (as well as the milestone report), helped write documentation for the model and experiment with it, helped create the goal of the project, and helped plan how we were going to tackle that goal. Andrés helped write some of the data preprocessing code, reviewed and edited the final report (and milestone report), wrote the project proposal, wrote a lot of the code for the model and experimented with it, helped create the goal of the project, and source it (do research), and helped plan how we were going to accomplish that goal.

8. References

Arthur Toth et al. (2017), *Predicting Shopping Behavior with Mixture of RNNs,* SIGIR 2017 eCom

Jongsung Kim (2022), *Development of a Deep Learning-Based Prediction Model for Water Consumption at the Household Level,* 14 Water 1512.

Kiran Chaudhary et al.(2021), *Machine learning-based mathematical modelling for prediction of social media consumer behavior using big data analytics,* Journal of Big Data

Nithin Soundar et al., *Retail Demand Forecasting using CNN-LSTM model,* Proceedings of the International Conference on Electronics and Renewable Systems.

Priyam Saha et al.(2022), *Demand Forecasting of a Multinational Retail Company using Deep Learning Frameworks,* 55 IFAC PapersOnLine

Rashed Ibrahim (2022), *Buyer Prediction Through Machine Learning,* Rochester Institute of Technology

Sushil Punia et al.(2020), *Deep learning with long short-term memory networks and random forest for demand forecasting in multi-channel retail,* 58 International Journal of Production Research

Zeynep Hilal Kilimci et al.(2019), *An Improved Demand Forecasting Model Using Deep Learning Approach and Proposed Decision Integration Strategy for Supply Chain,* Complexity

9. Appendix

Binary crossentropy

```
Test loss: 0.7082814574241638
Test accuracy: 0.024971907259896398
Test AUROC: 0.49339550733566284
Test precision: 25.0
Test recall: 0.14306152006611228
```

Loss No. 1

```
Test loss: -9481.2626953125
Test accuracy: 26.183044910430908
Test AUROC: 0.5
Test precision: 26.183044910430908
Test recall: 100.0
```

Loss No. 2

```
Test loss: -199.22288513183594
Test accuracy: 26.183044910430908
Test AUROC: 0.5
Test precision: 26.183044910430908
Test recall: 100.0
```

Loss No. 3

```
Test loss: 0.0
Test accuracy: 73.81695508956909
Test AUROC: 0.5
Test precision: 0.0
Test recall: 0.0
```

Loss No. 4

```
Test loss: 0.25
Test accuracy: 26.183044910430908
Test AUROC: 0.5
Test precision: 26.183044910430908
Test recall: 100.0
```

Loss No. 5

```
Test loss: 0.25
Test accuracy: 26.183044910430908
Test AUROC: 0.5
Test precision: 26.183044910430908
Test recall: 100.0
```