Ryerson University
Department of Electrical
& Computer Engineering
Faculty of Engineering & Architectural Science

Semester/Year (e.g. F2017)	Fall 2021

Instructor	Dr. Ebrahim Bagheri
Graduate Assistant	Amin Bigdeli

Assignment:	Bank Customer Churn Prediction

Submission Date:	November 28, 2021
Due Date:	December 3, 2021

Student LAST Name	Student FIRST Name	Student Number	Signature*
Sivaloganathan	Janakan	500960836	

*By signing above you attest that you have contributed to this written lab report and confirm that all work you have contributed to this lab report is your own work. Any suspicion of copying or plagiarism in this work will result in an investigation of Academic Misconduct and may result in a "0" on the work, an "F" in the course, or possibly more severe penalties, as well as a Disciplinary Notice on your academic record under the Student Code of Academic Conduct, which can be found online at: http://www.ryerson.ca/senate/current/pol60.pdf

Abstract

The purpose of this experiment was to predict the likelihood of a customer leaving a banking institution based upon a provided set of customer features. ANN (Artificial Neural Networks) and classic machine learning algorithms such as decision tree and support vector machines were used to train the models as discrete input and output labels were provided for the customer dataset. After performing the required tasks on the customer bank churn dataset, herein lies my assignment report.

Introduction

To determine the characteristics of the churning model, the metrics were evaluated upon accuracy and output error. Through a sequence of trials and model building, this laboratory experiment will be deemed successful if the models will be able to predict customer attrition.

Methods and Materials

The materials that were used to conduct the experiment are as follows:
Interpreted language: Python3
Libraries: PyTorch, NumPy, Pandas, Scikit-Learn, Matplotlib, Seaborn

Experimental Procedure

The experiment was sequenced through data collection, validation, and cleaning to ensure that the dataset that was fed into the model did not display any errors which would skew results. The *Excited* column was utilized as the independent variable with the inclusion of the customer-related banking information such as credit score.

The customer churn model was developed from the following through sequencing of the following procedures:

1. Data Preprocessing
2. Data Evaluation
3. Model Selection
4. Model Evaluation
5. Model Improvement
6. Future Predictions
7. Model Deployment

Data Exploration

The data that was inputted into the model is represented in Figure 1. This model was validated and cleaned so anomalies can be rid of. Only one row was deleted due to an outlying value for the target value as *Ω*. The *RowNumber*, *CustomerId*, and *Surname* were excluded in the analysis as it is assumed that there is no relationship between these independent variables and the dependant variable *Exited*. Aside from the one outlier, other attributes, duplications, and empty records were unidentified, thus presenting a modelable set.

Through feature extraction, the standardized variable that portrays meaningful value are as follows: CreditScore, Geography, Gender, Age, Tenure, Balance, NumOfProducts, HasCrCard, IsActiveMember, EstimatedSalary, *Exited*. Three pivot tables were utilized to explore the correlation between the data and the target variable. The three investigations that were conducted were on the independent variables of Gender, Geography, and Excited. Due to the small size of the dataset, Microsoft Excel was utilized to conduct the data exploration. If the dataset were to be of a large size, then the operations would have been performed using Python scripts due to its speed in processing and versatility in wrangling the data. The sum was depicted for columns that were categorized by either 1 or 0 thus, a total number of 1 of a column would showcase the sum of true statements for that label. As the data provided presented categorical data rather than continuous, explicit pivot tables were utilized to showcase correlations amongst the variables of interest.

The extraneous variables were identified to ensure consistency throughout the dataset. Fortunately, from an observation of key variables of interest such as credit card, credit score, age, and estimated salary, no such outliers were identified. This brief exploration concludes that the machine learning models would be trained based on an accurate and consistent dataset.

Row Labels	Sum of Exited	Sum of IsActiveMember	Average of EstimatedSalary	Sum of HasCrCard	Average of NumOfProducts	Average of Balance	Average of Tenure	Average of Age	Average of CreditScore
France	**810**	**2591**	**99899.90464**	**3542**	**1.530819868**	**62105.02284**	**5.004588071**	**38.51166966**	**649.6441253**
Female	460	1162	99564.25276	1578	1.547545334	60322.67016	4.950022114	38.77399381	649.1857585
Male	350	1429	100175.671	1964	1.517078488	63569.37582	5.049418605	38.29614826	650.0207122
Germany	**814**	**1248**	**101113.4351**	**1791**	**1.519728976**	**119730.1161**	**5.009964129**	**39.77162216**	**651.4535672**
Female	448	559	102446.4241	843	1.51131601	119145.9665	4.965632858	40.15423303	653.093881
Male	366	689	99905.03396	948	1.527355623	120259.6682	5.050151976	39.42477204	649.9665653
Spain	**413**	**1312**	**99440.57228**	**1721**	**1.539362132**	**61818.14776**	**5.032297134**	**38.89099717**	**651.3338716**
Female	231	563	100734.1075	771	1.573002755	59862.09253	5	39.19926538	651.7695133
Male	182	749	98425.68768	950	1.5129683	63352.83375	5.057636888	38.64913545	650.9920749
Grand Total	**2037**	**5151**	**100090.6219**	**7054**	**1.530153015**	**76493.53864**	**5.01280128**	**38.92179218**	**650.5167517**

Figure 1.0. Statistics based upon country categorized by gender

Row Labels	Sum of Exited	Sum of IsActiveMember	Average of EstimatedSalary	Sum of HasCrCard	Average of NumOfProducts	Average of Balance	Average of Tenure	Average of Age	Average of CreditScore
Female	1139	2284	100601.5414	3192	1.544133832	75659.36914	4.966101695	39.23838873	650.831389
Male	898	2867	99665.19899	3862	1.51851173	77188.1193	5.051686217	38.65817449	650.2547654
Grand Total	**2037**	**5151**	**100090.6219**	**7054**	**1.530153015**	**76493.53864**	**5.01280128**	**38.92179218**	**650.5167517**

Figure 1.1. Statistics based upon gender

Row Labels	Sum of IsActiveMember	Average of EstimatedSalary	Sum of HasCrCard	Average of NumOfProducts	Average of Balance	Average of Tenure	Average of Age	Average of CreditScore
0	**4416**	**99738.82731**	**5630**	**1.544209997**	**72754.43334**	**5.033283095**	**37.4081889**	**651.8382316**
Female	**1870**	**99816.07149**	**2397**	**1.555229142**	**71183.24964**	**4.977085781**	**37.38249119**	**652.1524677**
France	990	98526.82063	1257	1.556912826	58424.31006	4.957245974	37.09883398	650.5363687
Germany	394	103987.0422	526	1.538255034	118828.5144	4.899328859	37.34362416	654.514094
Spain	486	98900.64909	614	1.566433566	56594.82069	5.086247086	38.01165501	653.4941725
Male	**2546**	**99681.13991**	**3233**	**1.535980693**	**73927.82283**	**5.075252304**	**37.42738043**	**651.6035542**
France	1303	99735.88784	1716	1.529558701	61800.2217	5.041631973	37.33805162	651.5815987
Germany	558	101181.5775	688	1.561052632	119896.5291	5.1	37.28631579	652.2789474
Spain	685	98390.16126	829	1.529021559	61871.62629	5.122719735	37.71641791	651.115257
1	**735**	**101465.6775**	**1424**	**1.47520864**	**91108.53934**	**4.932744232**	**44.83799705**	**645.3514973**
Female	**414**	**102948.9861**	**795**	**1.510974539**	**89036.63936**	**4.933274802**	**44.78489903**	**646.8832309**
France	172	103626.0251	321	1.510869565	67755.16263	4.92173913	45.3326087	643.8978261
Germany	165	99884.45873	317	1.466517857	119673.8723	5.075892857	44.828125	650.7321429
Spain	77	107544.0958	157	1.597402597	71997.67368	4.67965368	43.61038961	645.3636364
Male	**321**	**99584.28727**	**629**	**1.429844098**	**93736.48374**	**4.932071269**	**44.90534521**	**643.408686**
France	126	103193.8397	248	1.431428571	75710.8278	5.102857143	44.87142857	639.3085714
Germany	131	96591.60126	260	1.43989071	121202.2424	4.920765027	44.97540984	643.9644809
Spain	64	98661.09901	121	1.406593407	73167.8678	4.626373626	44.82967033	650.1758242
Grand Total	**5151**	**100090.6219**	**7054**	**1.530153015**	**76493.53864**	**5.01280128**	**38.92179218**	**650.5167517**

Figure 1.2. Statistics based upon target value categorized by gender and geography

Row Labels	Sum of IsActiveMember	Average of EstimatedSalary	Sum of HasCrCard	Average of NumOfProducts	Average of Balance	Average of Tenure	Average of Age	Average of CreditScore
0	**4416**	**99738.82731**	**5630**	**1.544209997**	**72754.43334**	**5.033283095**	**37.4081889**	**651.8382316**
France	2293	99217.79837	2973	1.541280038	60353.63192	5.005472282	37.23554604	651.133714
Germany	952	102414.6579	1214	1.551032448	119427.1067	5.01179941	37.31150442	653.2613569
Spain	1171	98602.36986	1443	1.544573643	59678.07047	5.10755814	37.83914729	652.1041667
1	**735**	**101465.6775**	**1424**	**1.47520864**	**91108.53934**	**4.932744232**	**44.83799705**	**645.3514973**
France	298	103439.2783	569	1.47654321	71192.79573	5	45.13333333	641.9148148
Germany	296	98403.88645	577	1.454545455	120361.0756	5.006142506	44.89434889	647.6891892
Spain	141	103629.5548	278	1.513317191	72513.35245	4.656174334	44.14769976	647.4842615
Grand Total	**5151**	**100090.6219**	**7054**	**1.530153015**	**76493.53864**	**5.01280128**	**38.92179218**	**650.5167517**

Figure 1.3. Statistics based upon target value categorized by geography

Extraneous records

RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProdu	HasCrCard	IsActiveMem	EstimatedSa	Exited
6647	15662021	Lucciano	685	Spain	Female	42	2	0	2	0	0	199992.48	0

Figure 1.4. Customer with a highest estimated salary

RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProdu	HasCrCard	IsActiveMem	EstimatedSa	Exited
2363	15791053	Lucciano	709	Germany	Male	45	4	122917.71	1	1	1	11.58	1

Figure 1.5. Customer with a lowest estimated salary

RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProdu	HasCrCard	IsActiveMem	EstimatedSa	Exited
6760	15660878	T'ien	705	France	Male	92	1	126076.24	2	1	1	34436.83	0

Figure 1.6. Oldest customer

RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProdu	HasCrCard	IsActiveMem	EstimatedSa	Exited
7957	15731569	Hudson	850	France	Male	81	5	0	2	1	1	44827.47	0

Figure 1.7. Customer with highest credit score

RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProdu	HasCrCard	IsActiveMem	EstimatedSa	Exited
8763	15765173	Lin	350	France	Female	60	3	0	1	0	0	113796.15	1

Figure 1.8. Customer with lowest credit score

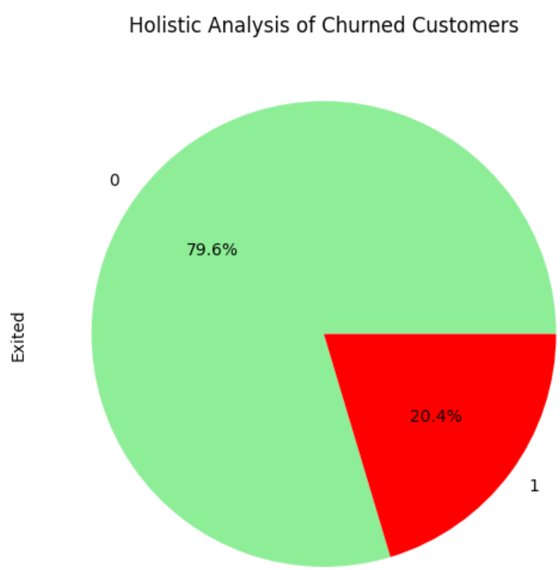

Figure 2.0. Customer churn representation

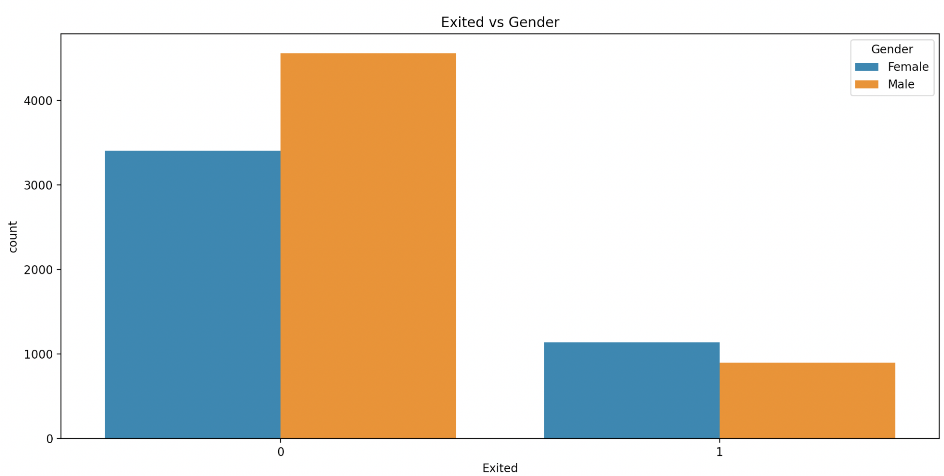

Figure 2.1. Customer churn and gender representation

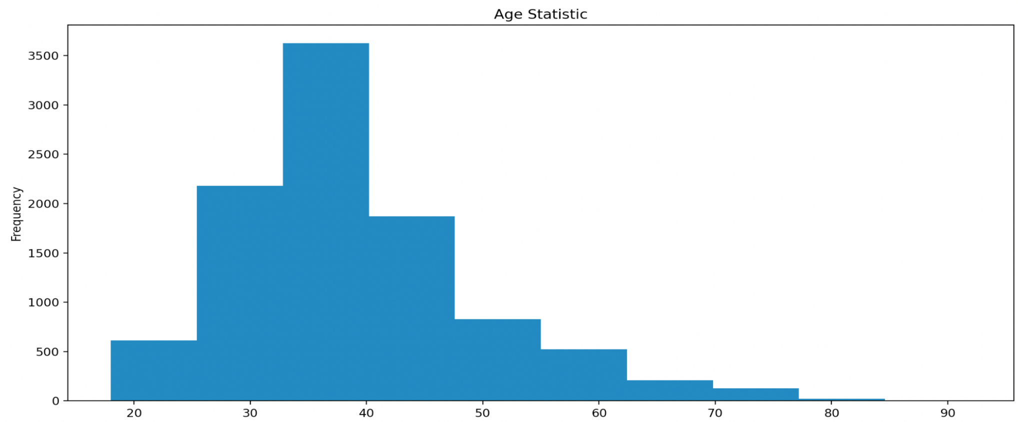

Figure 2.2. Population distribution of age

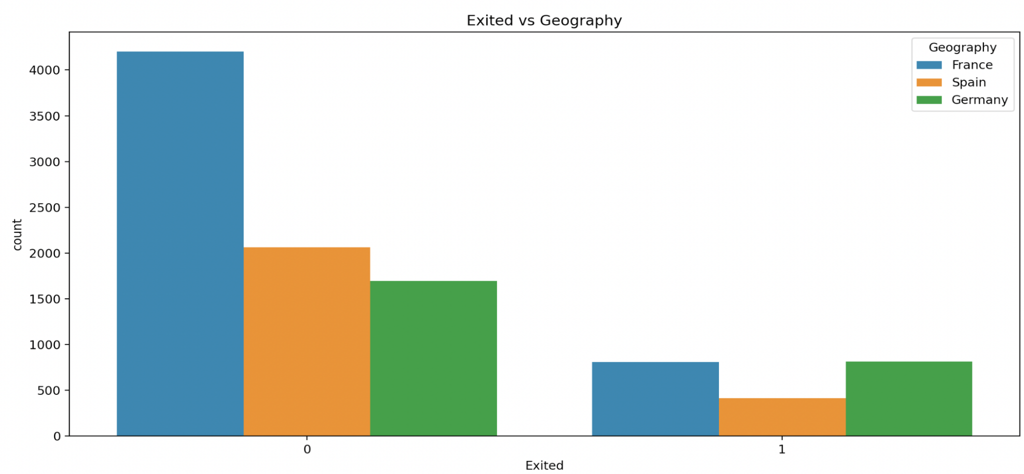

Figure 2.3. Customer churn and geography comparison

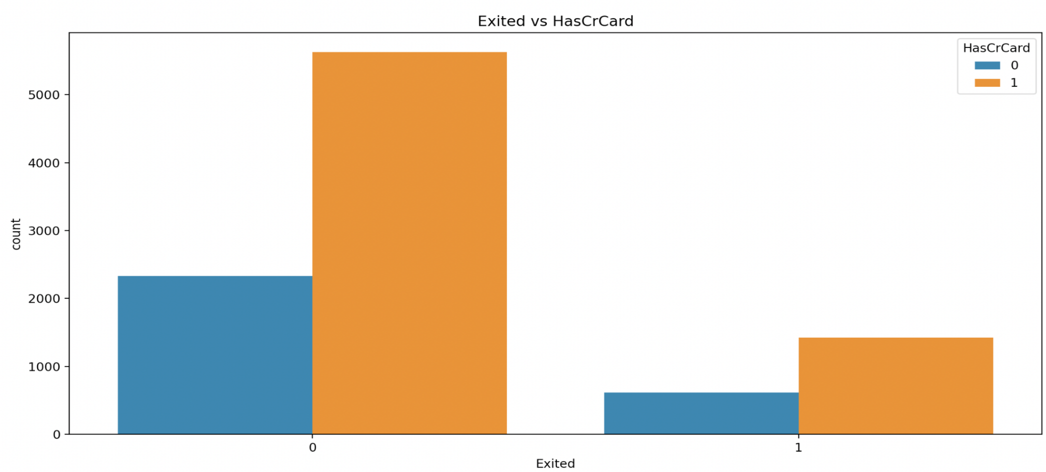

Figure 2.4. Customer churn and credit card ownership comparison

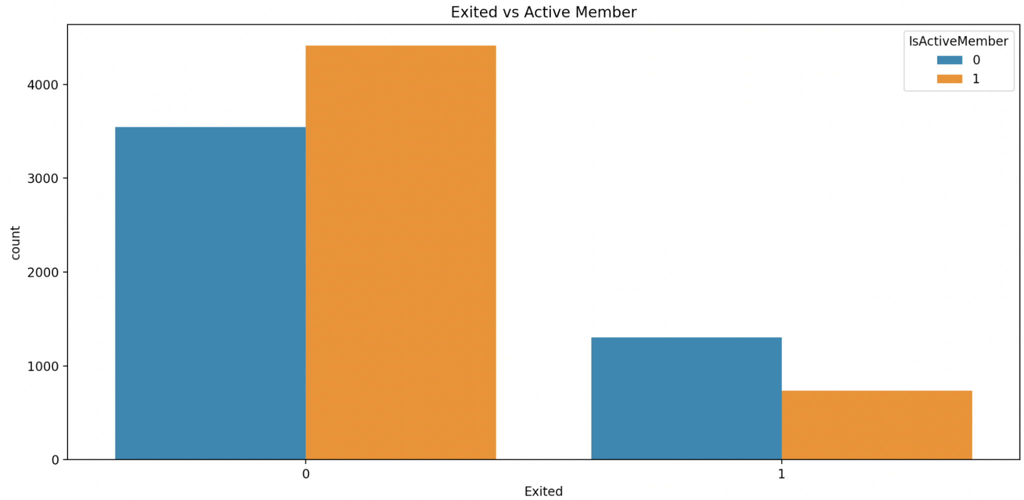

Figure 2.5. Customer churn and active member comparison

From the exploration of the data, it is clear that the majority of the data were from the France geographical location. Additionally, the majority of the customers who have churned possess a credit card which may present some insight as to if there is a relationship between the bank's credit card offers compared to others. The relationship between the gender and the target variable showcases that female customers are more likely to churn compared to male customers. Finally, the comparison among the inactivity and target indicates that inactive customers are more likely to leave the banking institution. Therefore, from an analysis of the data, there are indications that the bank requires a redesign of credit card promotions, greater expansion to other geographical locations, services to prevent customer inactivity, and possibly a gender-inclusive program to intrigue female customers.

Results

The customer churn prediction resulted in a successful output based upon the information that was provided to the customer. In order to evaluate the results, the accuracy was compared among 3 test case scenarios. The test cases were also provided with a specific input to determine the forecasted target value for verification.

Performance of the classic models

Classification accuracy of the decision tree model=

$$\frac{(93.84 + 93.7 + 93.65)}{3} = 93.73 \pm 0.0985\%$$

$$s = \sqrt{\frac{\Sigma(x-\overline{x})^2}{n-1}} = \sqrt{\frac{(93.84-93.73)^2 + (93.70-93.73)^2 + (93.65-93.73)^2}{3-1}} = 0.0985$$

Classification of the support vector machine model=
$\frac{(79.06 + 78.93 + 80.03)}{3} = 79.34 \pm 0.6011\%$

$$s = \sqrt{\frac{\Sigma(x-\bar{x})^2}{n-1}} = \sqrt{\frac{(79.06-79.34)^2 + (78.93-79.34)^2 + (80.03-79.34)^2}{3-1}} = 0.6011$$

Performance of the deep learning model with the following parameters:
Epochs= 1000 by 100 iterations
Hidden Neurons= 15+8= 23
Hidden layers= 2
Activation function= Sigmoid
Optimizer function= Adam
Classification of the neural network model= $\frac{(84.36 + 84.71 + 83.86)}{3} = 84.31 \pm 0.4272\%$

$$s = \sqrt{\frac{\Sigma(x-\bar{x})^2}{n-1}} = \sqrt{\frac{(84.36-84.31)^2 + (84.71-84.31)^2 + (83.86-84.31)^2}{3-1}} = 0.4272$$

This assignment required classification models instead of regression models because we wanted to predict discrete class labels.

Performance Analysis

The performance was determined by the definition of accuracy which was printed to the standard output after running the Python3 script. The presented accuracy showcases the number of correct predictions over the total number of predictions made. The percentage of accuracy for both classic machine learning models and neural network models has been retrieved after 3 instances of repeated output.

Output error for classic models
Error percentage for decision tree model= $100.0 - 79.34 = 6.270 \pm 0.0985\%$
Error percentage for support vector machine model=
$100.0 - 79.34 = 20.66 \pm 0.6011\%$

Output error for deep learning model
Error percentage for neural network model= $100.0 - 84.31 = 15.69 \pm 0.4272\%$

The accuracy and the error propagated from the output of the trained machine learning models must sum to the probability totality of 1, which is also 100%. As the accuracy of the output was determined from the models, the output error was determined by finding the difference of 1 and the accuracy decimal. The output error is calculated below.

Parameter Sensitivity Analysis

The parameter sensitivity analysis was determined by the accuracy of several trial experiments. In order to be deemed successful in optimizing the accuracy of the deep learning model, the default hyperparameters were set to prevent overfitting and underfitting.

In order to determine the effect of the variance of epoch values, a graphical model as presented below was produced to showcase the iterations of loss and accuracy values per epoch value. This graph displays similar behavior to a negative exponential function as the loss values exponentially decrease and stagnates at higher epoch inputs. The default parameters that were used were: iterations of 200 epoch to maximum 5000, Adam optimizer function, sigmoid activation function, a total of 18+5 hidden neurons, and 2 hidden layers. Additionally, the accuracy values with increasing epochs also decrease.

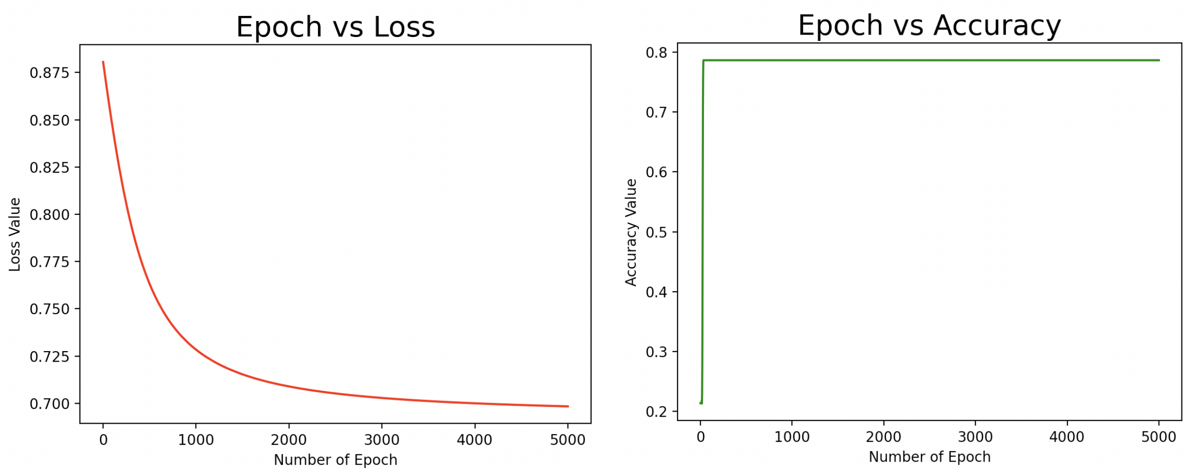

Figure 3.0. Performance of epoch intervals

To determine the parameter characteristics of the other variables, a total of 6 trials were completed. The trials were categorized by the two optimizer functions Adam and SGD and then subcategorized by the hyperbolic tan function, rectified linear unit, and sigmoid activation functions. The default 23 (15+8) hidden neurons, 2 hidden layers, and

Adam optimizer function

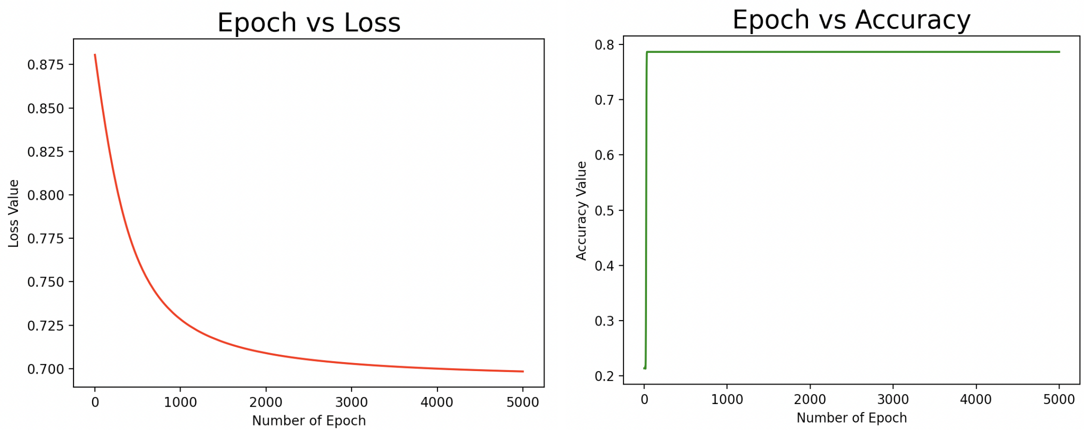

Figure 4.0. Performance of sigmoid function on default adam optimizer

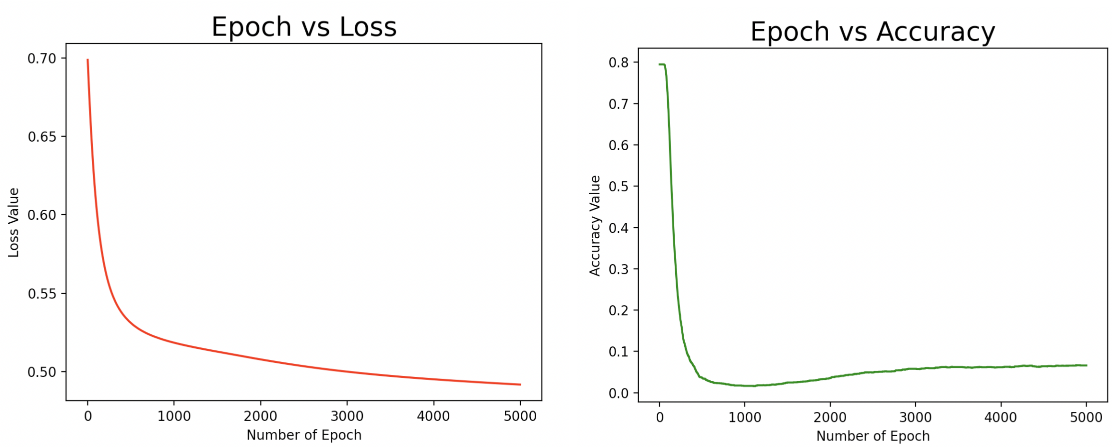

Figure 4.1. Performance of hyperbolic tangential function on default adam optimizer

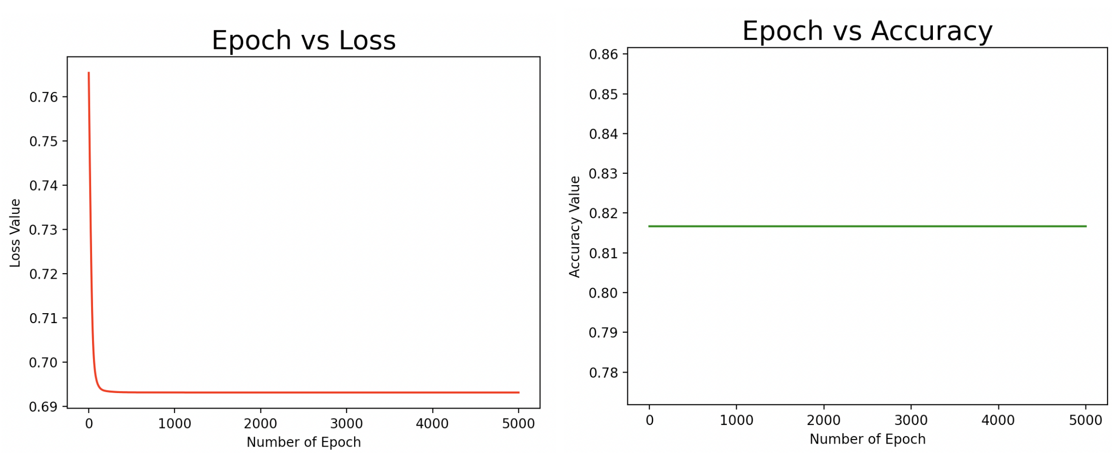

Figure 4.2. Performance of rectified linear unit function on default adam optimizer

SGD (Stochastic Gradient Descent optimizer function)

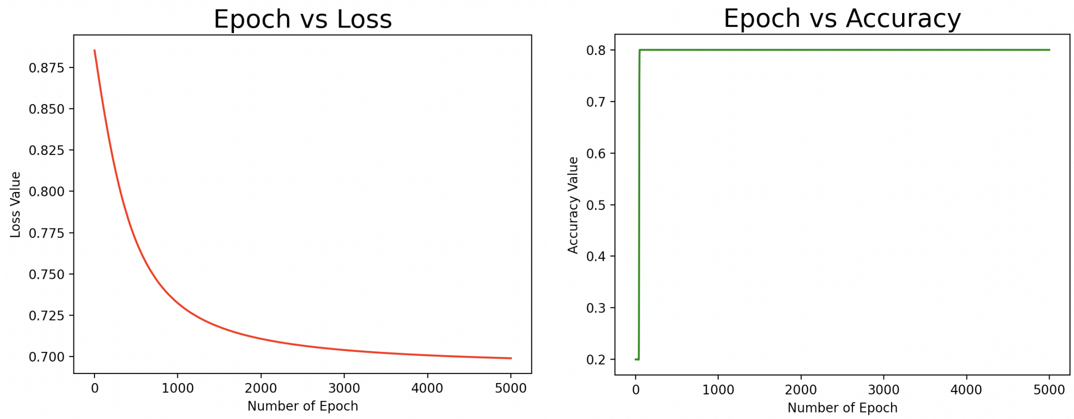

Figure 5.0. Performance of sigmoid function on default SGD optimizer

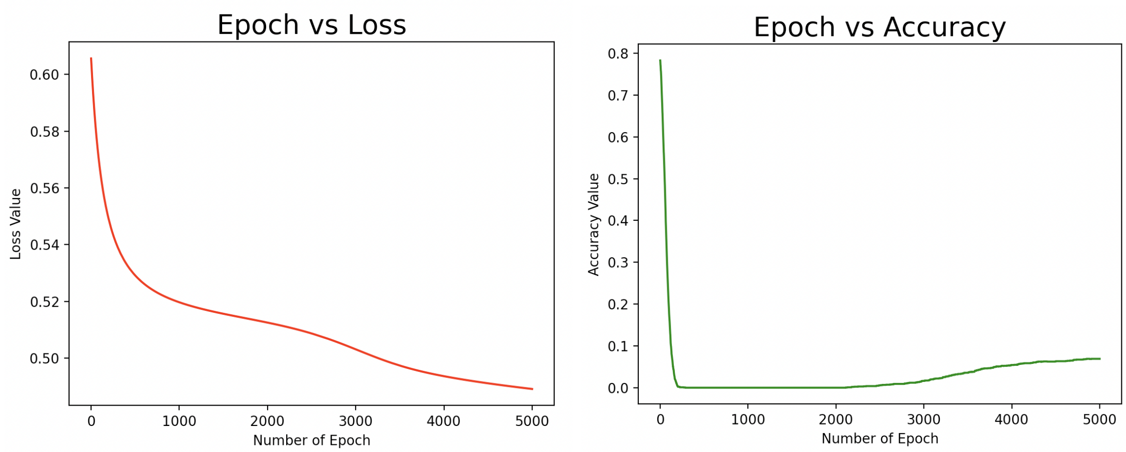

Figure 5.1. Performance of hyperbolic tangential function on default SGD optimizer

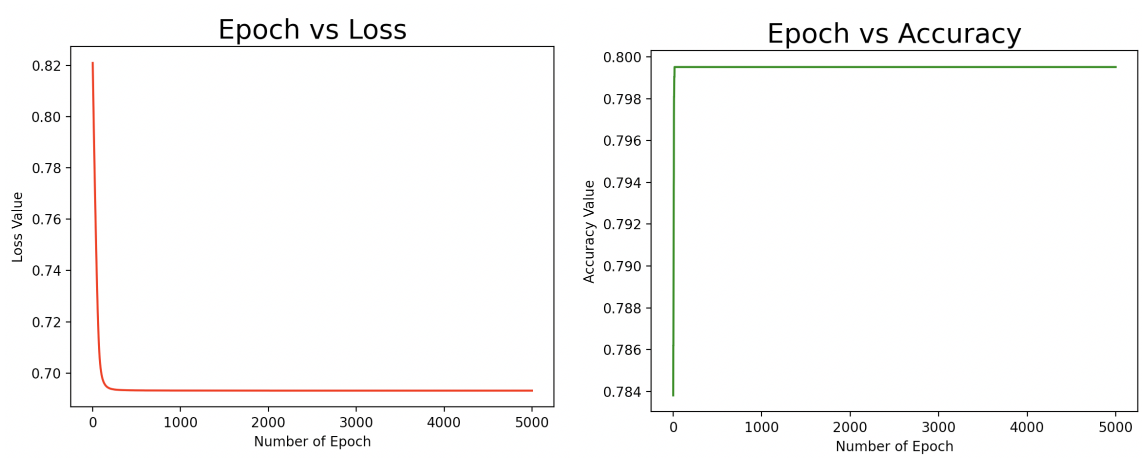

Figure 5.2. Performance of rectified linear unit function on default SGD optimizer

To test the change in neurons and hidden layers, the default values were used as follows:
Epochs= 5000 by 200 iterations
Activation function= Sigmoid
Optimizer function= Adam

Altering neuron quantity

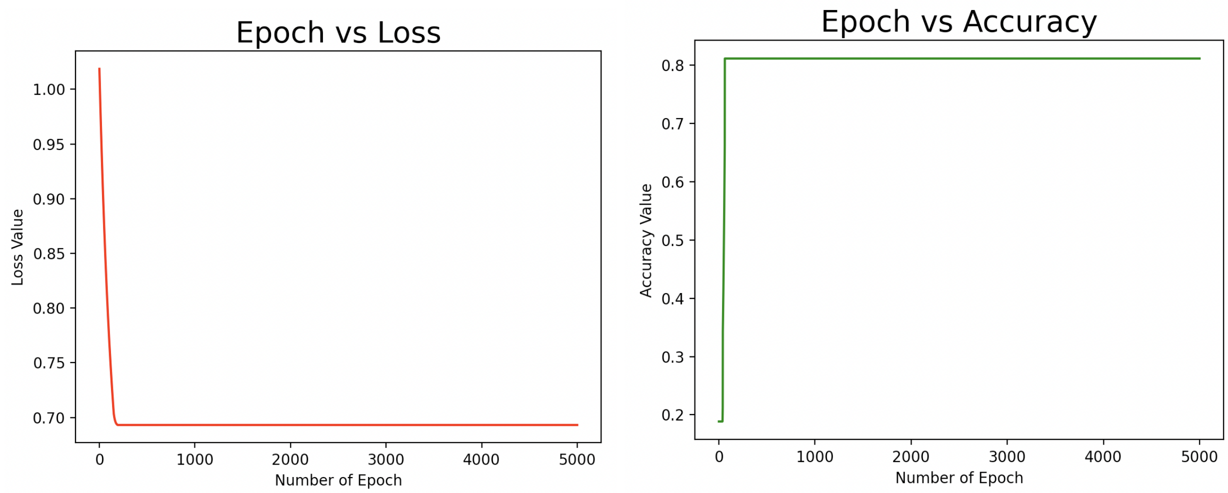

Figure 6.0. Performance of decreasing the number of hidden neurons (2+2)

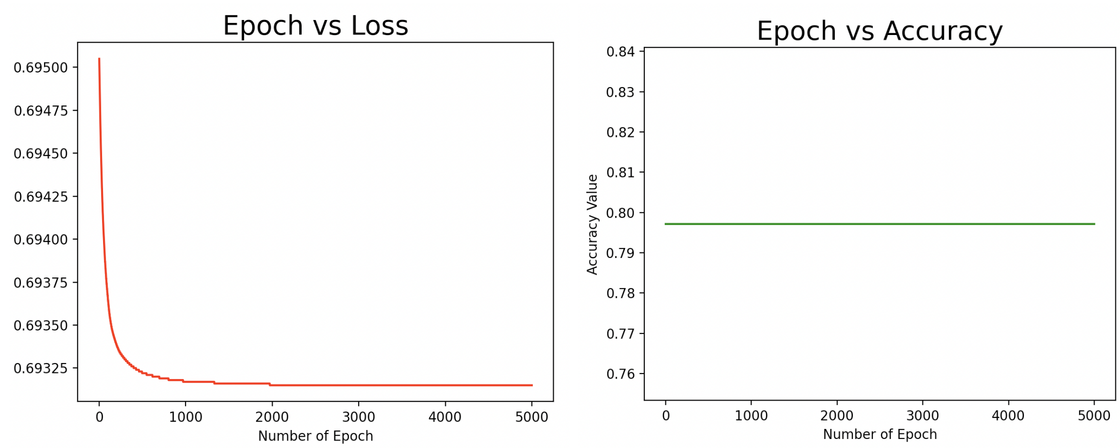

Figure 6.1. Performance of increasing the number of hidden neurons (20+10)

Altering hidden layer quantity

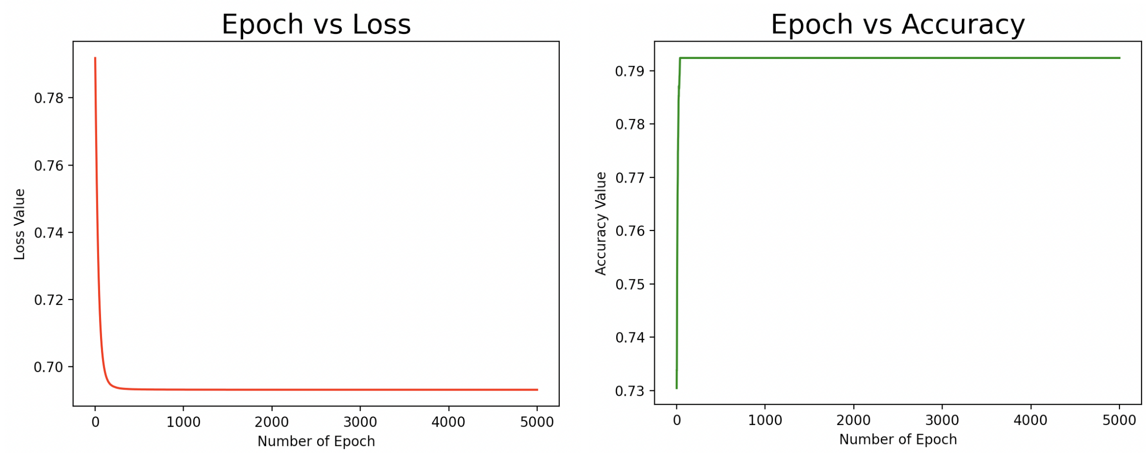

Figure 7.0. Performance of decreasing the number of hidden layers to 1

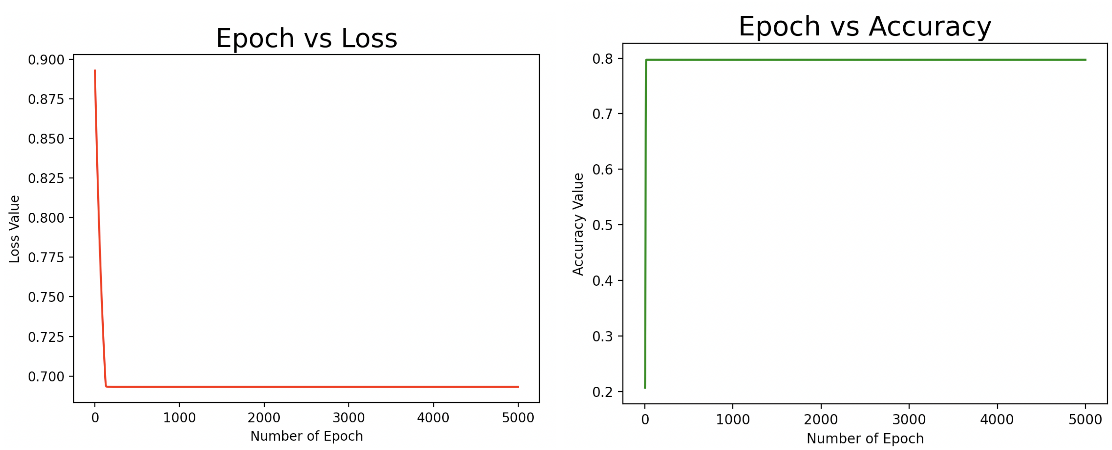

Figure 7.1. Performance of Increasing the number of hidden layers to 3

From the experimental results, it was evident that as the number of processing has increased (epochs), the loss function decreased substantially eventually stagnating to a certain point. From the variations of the activation functions, hyperbolic tangential presented with the worst results displaying accuracies to around 10% as the number of epochs increased. The rectified linear unit presented the most desirable results for both optimization functions which is expected as the interval for input is unrestricted with the increase of inputs as compared to the sigmoid and hyperbolic tangential functions. The adam function presented the most desirable results as the rectified linear unit produced an accuracy of 81.6% compared to 79.7% for the stochastic gradient descent function. Both optimizer functions showcased similarities with a few fluctuations in regards to the hyperbolic tangential function due to the properties of the function.

The parameters for experimentation for hidden layers and neurons were proceeded with the adam optimizer function and rectified linear unit activation function as it proposed the best results for the provided dataset. Although no drastic changes were displayed amongst the increase and decrease of neurons, the overall altering of the neurons did deteriorate the accuracy and present an increase in the loss value. The reason this has occurred is due to overfitting or underfitting as it requires several iterations to determine the most optimal accuracy rating. The initial accuracy of the model was 81.6%, but the increase in neurons provided an accuracy of 79.7% and the decrease in neurons provided 80.1%. The altering of hidden layers also proposed a deterioration inaccuracy which is predicted as neurons will present greater loss progressing through one layer or many layers. This also showcases the possibility of overfitting and underfitting as each iteration will either process the input values through too many layers skewing the data or providing scarce results from only one layer progression.

Discussions

From the exploration of the data through Microsoft Excel, it was evident that the age of the churned customers was of the average age of 45 while the average age of the non *Exited* customers was averaged to 37. This chart also showcases that the average balance of the customers who *Exited* is greater than those who have not *Exited*. Additionally, it is clear to see a relationship between the number of customers who have a credit card and their *Exited* value, for example, Spain has the least amount of *Exited*, but the sum of credit cards of that geographic population is almost 50% less than France and Germany. Based upon such conclusions, the prediction from the machine learning models was verified.

Machine learning algorithms are leverage structured, labeled data to make predictions, so the input data for the model are defined and organized into tables. Machine learning requires the human to label the data accurately as opposed to deep learning. Although the neural network was not optimal in this experiment compared to the other models, it is acknowledged that the effectiveness of deep learning exponentially increases with the increased volume and complexity of a dataset.

Analysis of epoch parameters
The number of times a whole dataset is passed through the neural network model is called an epoch. Too few epochs present underfitting since the neural network has not learned enough, while too many epochs will lead to overfitting.

Analysis of neurons
Since neurons represent the features that are being processed by the input data, utilizing fewer neurons will decrease the accuracy of the model since it will not account

for particular weights. However, if too many neurons were placed this will present overfitting which is due to the data being processed by more features than required for an optimal prediction. Therefore, in order to determine the right amount of neurons it is recommended to experiment and try to cover a little above the total amount of features in the dataset. The number of neurons will vary per situation, but for this classification experiment, utilizing one neuron for output and 15+8 hidden neurons was sufficient through trial and error.

Analysis of hidden layers
Through the experimentation, it was justified to maintain from 1-5 hidden layers to develop a functional model. Additionally, this experiment showcases that the most effective way to configure the neurons is to decrease the amount through each layer. This is deemed optimal because a large number of neurons in the first layer will learn a great number of fundamental features and then learn higher-level features in the layers to be introduced. This configuration was why the artificial neural network was set to have 15 neurons in the first hidden layer and 8 neurons for the second layer as it provided better performance.

Analysis of the sigmoid, hyperbolic tan, and rectified linear activation function
The activation function decides how to compute the input values of a layer into output values. This function is critical to the processing of complex tasks such as language translation and image classification. The sigmoid function normalizes the outputs on an interval between 0 and 1 based on the input values. This function is also known as the logistic function. The function is defined by $y = \frac{1}{1+e^{(-x)}}$. The hyperbolic tangent activation function provides a wider range of -1 to 1 output based on the specified input and it is represented as $y = tanh(x)$. The hyperbolic tangent function is able to accept negative values to -1 whereas the sigmoid function is only capable of accepting values greater than 0, due to this characteristic the hyperbolic tangential function is known as the zero-centered activation function. the rectified linear unit activation function scales the input variable linearly and cancels the values below 0. This function depletes negative values and proves an unrestricted capacity for the output. This feature allows maximum efficiency and speed when processing inputs.

Analysis of stochastic gradient descent function and adam optimizer function
The optimizer is responsible to change the learning rate and the weights of neurons in the neural network to reach the minimum loss function. This function is very important to achieve the highest accuracy and minimum loss. The learning rate controls the step size for a model to reach the minimum loss function. With the lower learning rate, you would require higher epochs and resources, but a higher learning rate will compute the

model faster. The choice of binary cross-entropy with logits loss function as the model that needed to be developed as a classification problem.

Conclusion

In conclusion, this experiment demonstrates how careful adjustments to parameters can greatly and the identification of the classifier will impact the output result. Since the dataset is composed of customer information on banking, the appropriate classifier was utilized. The most optimal classification model to predict the customer churn information would be the decision tree due to its high accuracy of $93.73 \pm 0.0985\%$ and low output of $6.270 \pm 0.0985\%$. This is justified because decision trees derive discrete labels or categories more effectively as support vector machine presents an advantage to non-linear problems. This experiment covers a holistic view of the classical machine learning models and artificial neural networks. This research showcases how important the hyperparameters, model type, data processing, and experimentations are when developing a recommendation system.

This research did not cover many variables such as the learning function as it was set statically to 0.01 and the binary cross-entropy with a logits loss function. Additionally, the performance of the model was not timed as well. As next steps, it is recommended to experiment with more diverse test cases per hyperparameter and understand if there is any correlation between learning function and activation function or if the altering of activation function and optimization function alter the timing to train.

References

Agrawal, T., & Safari, an O'Reilly Media Company. (2020). *Hyperparameter optimization in machine learning: Make your machine learning and deep learning models more efficient (1st ed.)*. Apress.

Amuda, K. A., & Adeyemo, A. B. (2019). Customers churn prediction in financial institution using artificial neural network.

Chintala, S. C. (n.d.). *Deep Learning with PyTorch: A 60 Minute Blitz — PyTorch Tutorials 1.10.0+cu102 documentation*. PyTorch. Retrieved December 1, 2021, from https://pytorch.org/tutorials/beginner/deep_learning_60min_blitz.html

Fridrich, M. (2017). Hyperparameter optimization of artificial neural network in customer churn prediction using genetic algorithm. Trendy Ekonomiky a Managementu, 11(28), 9-21. https://doi.org/10.13164/trends.2017.28.9

Livshin, I., & Safari, an O'Reilly Media Company. (2019). Artificial neural networks with java: Tools for building neural network applications (1st ed.). Apress.

Ryerson University, & Bigdeli, A. B. (2021, November). Bank Customer Churn Prediction. Amin Bigdeli.

Shreemali, J., Chakrabarti, P., Poddar, S., & Kothari, N. (2021). Applying different machine learning classifiers on very large dataset to demonstrate the challenges in choosing the right machine learning classifier for a given problem. *Materials Today : Proceedings,* https://doi.org/10.1016/j.matpr.2020.12.140