Application of Neural Network with Innovative Loss in Classification Problem

Zibo Tian

Introduction

As population aging has become a worldwide problem, and over-65 population in the U.S. will more than double over the next 40 years according to the Urban Institute, there will be increasing demand for home healthcare services, which is a modern-developed approach especially beneficial for the old [1]. The Home Care Providers industry is among the fastest-growing healthcare industries in the United States [2]. Given the high industry growth with a huge number of emerging agencies, competitive newcomers are expected to have a better quality of services provided. In this extended abstract, a classification model will be discussed based on the Home Health Quality Reporting Program Data on Care Compare Version 4.0 (HHCAHPS) [3], which is a recently updated data set related to the nationwide home health care agencies. The main outcome of interest is the star-rating awarded by the program that comprehensively evaluates the quality of service provided by the corresponding home health care agencies. In addition to the routine application of the multinomial logistic regression model, a dense neural network was used to enhance the accuracy of classification. With well-trained models that appropriately predict the star-rating of the target agencies based on their conditions on all aspects, management teams of the existing home health care agencies or potential stakeholders of the new agencies can use hypothetical data to enhance their operating strategies and better understanding the market demand.

Theoretically, the logistic regression can be viewed as a simplified neural network with a single layer. This project mainly focused on model training (weight tunning) process within the deep neural network approach. When considering the choice of loss function used in the back-propagation of error process, in spite of the categorical cross-entropy, I innovatively tried a weighted cross-entropy and a penalized cross-entropy with defined penalty weight. Regularization by means of the dropout and L2 regularization approaches were conducted to suppress the overfitted weights. Cross-validation within the training data set was implemented to search optimized dropout probabilities and the λ parameter in L2 regularization respectively. Under this problem with many types of the categorical outcome and a complicated set of

covariates, a well-trained dense neural network is expected to did better job than the logistic regression.

Method

The data set used in this analysis is the most recent version of the Home Health Quality Reporting Program Data on Care Compare. Home health care agencies are the unit of observations. With the goal of constructing a classification model that potentially predict the agencies' service quality, the star rating provided by the program was regarded as the categorical outcome variable with 7 levels (less or equal to 2, 2.5, 3, 3.5, 4, 4.5, and 5). For simplicity, all rows with missing data of the star rating were excluded. This procedure made this project a complete-case analysis, because any missing covariates induce the absence of the outcome variable by the nature of the original data. 4019 observations were included. After removing variables for identification such as name, address, etc., 29 covariates including 3 categorical variables with more than 2 levels, 4 binary variables, and 22 continuous variables were included in the cleaned data set for further analyses. All continuous covariates are scaled (standardized). A list of included covariates is provided in the appendix.

The data was evenly partitioned as training data (2010 observations) and test data (2009 observations). Each observation was randomly assigned to one of the groups. Fitting and tuning of the parameters were only conducted within the training data. Test data was only used for the final check and evaluation of the prediction accuracy based on adjusted models.

A multinomial logistic regression was firstly fitted using a logit link function with star rating as the outcome variable and all other variables as covariates. No interaction term, higher-ordered term, or spline terms were included in the model. Since the project mainly focuses on the analysis related to the application of the deep neural network, the fitted logistic regression model as well as its prediction accuracy would be used as a comparator that reflects the well-being of the neural network.

Based on data and the research question, the neural network has 36 inputs at the input layer and 7 outputs at the output layer. A grid search together with 7:3 hold-out validation within the training data helped to configure the appropriate number of layers and nodes in each layer. The

domain for the number of nodes was {32, 64, 128, 256} in the grid search, and for the number of hidden layers, {1, 2, 3, 4, 5}. The final decision was 2 hidden layers with 64 nodes in each. The validation process also indicated 10 as the proper number of epochs used before a series overfitting trend reflected by the decreasing accuracy as well as the increasing loss. The neural networks are trained using a modified stochastic gradient descent algorithm with 32-unit mini-batches (mini-batch gradient descent).

Given the categorical outcome of interest, a "softmax" activation function was used at the output layer. The loss function corresponding to the classification is the categorical cross-entropy. Enlightened by previous work [4] [5], I proposed two innovative loss functions that might be especially useful under this case with multiple levels of outcomes. The first one is a weighted categorical cross-entropy generalized from the weighted binary cross-entropy. The second one is a penalized categorical cross-entropy directed by a penalty matrix. Here show the formulas for each loss respectively.

$$f_1(y, \hat{y}) = -\sum_{i=1}^{7} y_i \times ln(\hat{y}_i)$$

$$f_2(y, \hat{y}) = -\sum_{i=1}^{7} w_i \times y_i \times ln(\hat{y}_i)$$

$$f_3(y, \hat{y}) = -\{\sum_{i=1}^{7} y_i \times ln(\hat{y}_i)\} \times Penalty_{|Order(y)-Order(\hat{y})|}$$

Among the equations above, y is the true type specified by a vector of length 7. For example, [0, 1, 0, 0, 0, 0, 0]' refers to the second type (2.5-star rating). $\hat{y}$ is the fitted probability vector with the current value, hence it has the same size as y. The weighted cross-entropy incorporates a weight vector that applies fixed weight to each type of the outcome. It can be used when the data is imbalanced on aspect of the types of the categorical variable. Intuitively, smaller loss is adjusted for more scarce types. When the weight vector has all elements equals to 1, the loss function f_2 would reduce to f_1. Several experiments were did based on different tentative weight vectors.

The penalized categorical cross-entropy is more complicated. A penalty matrix should be firstly introduced [5]. The matrix below defined the penalty term $Penalty_{|Order(y)-Order(\hat{y})|}$

for different pairs of y and $\hat{y}$. The $Order(y)$ notation means the ordinal of the one in vector y, and the $Order(\hat{y})$ means the ordinal of the largest element in vector $\hat{y}$. For example, given the penalty matrix as showed, if y = [0, 1, 0, 0, 0, 0, 0]' and $\hat{y}$ = [0.1, 0.1, 0.1, 0.1, 0.4, 0.1, 0.1]', $|Order(y) - Order(\hat{y})| = 3$, and $Penalty_{|Order(y)-Order(\hat{y})|} = 8$. Again, if y = [0, 1, 0, 0, 0, 0, 0] and $\hat{y} = [0.05, 0.05, 0.05, 0.7, 0.05, 0.05, 0.05]'$, $|Order(y) - Order(\hat{y})| = 2$, and $Penalty_{|Order(y)-Order(\hat{y})|} = 4$. The categorical star rating in this analysis is closed to an ordinal outcome. In the sense of the ordinal variable, the difference between quantified outcomes is no longer meaningless. Therefore, the intuition there is that we discriminate different types of misspecifications and try to avoid misspecifications that jump out of the acceptable spectrum. The numbers initialized in the matrix is just a tentative example, and they were used in the model training with the f_3 loss.

Order	Pred. type 1	Pred. type 2	Pred. type 3	Pred. type 4	Pred. type 5	Pred. type 6	Pred. type 7
True type 1	1	2	4	8	10	10	10
True type 2	2	1	2	4	8	10	10
True type 3	4	2	1	2	4	8	10
True type 4	8	4	2	1	2	4	8
True type 5	10	8	4	2	1	2	4
True type 6	10	10	8	4	2	1	2
True type 7	10	10	10	8	4	2	1

Figure 1: Defined Penalty Matrix. Type 1 refers to the less than 2 type, type 2 refers to the 2.5 type, type 3 refers to the 3 type, etc.

After specifying the loss function for the output layer, L2 regularization, and dropout method were used, in order to potentially avoid overfitting problems. Grid search together with a 5-fold cross-validation were used to search for an optimized λ from {0.004, 0.008, 0.012, …, 0.1}, and an optimized dropout probability p from {0.02, 0.04, …, 0.4}. The implementation of cross-validation was motivated by the concern of limited sample size in the training data. The 5-fold cross-validation process can be understood as the procedure followed. Within the training data, the observations are divided into 5 groups (folds). In the first iteration, the first fold is regarded as the test data, and the other four-fifth of the observations constitute the first training data. In the second iteration, the second fold becomes the test data and the left observations are viewed as the training data. After five iterations, the averaged accuracy of

prediction would be recorded and compared with other averaged accuracy corresponding to different values of λ or p initialized under the chosen approach. For simplicity, I did not do experiments on mixtures of different regularization methods within a single training process.

Results

As introduced in the method part, a grid search was used to determine the number of nodes and number of layers included in the neural network. The table below shows the prediction accuracies under the 7:3 holdout validation.

	32 nodes	64 nodes	128 nodes	256 nodes
1 hidden layer	0.633	0.670	0.672	0.668
2 hidden layers	0.654	0.689	0.679	0.677
3 hidden layers	0.652	0.682	0.682	0.603
4 hidden layers	0.660	0.671	0.620	0.552
5 hidden layers	0.644	0.675	0.498	0.530

Table 1: Grid Search. Accuracies over different combinations of nodes and layers.

The selected structure of the neural network with 2 hidden layers and 64 nodes in each layer did not give particularly outstanding performance, but this final choice basically complies with the recommendations [6]. I noticed that the prediction accuracy decreased sharply when the number of nodes in each layer becomes too large, which showed some evidence of overfitting given from models trained by complex (incompatible) neural networks [6].

A multinomial logistic regression was first fitted and tested. The accuracy of prediction computed under this regression method was used as a baseline comparator. Then, based on the logistics mentioned before, several neural network models were fitted. The figure below gives a review of the feature of each model. And the tables below show the prediction accuracies assessed by the test data set. M0 refers to the multinomial logistic regression model. M1 refers to the neural network trained model with the original cross-entropy loss. M2.1 and M2.2 represent the neural network model with f_2 and f_3 loss respectively. M3.0, M3.1, and M3.2 represent the models adjusted by the two regularization approaches. Since l2 regularization and the dropout method are separately implemented, each specific model actually has two branches.

For instance, M3.0 includes M3.0.1 and M3.0.2 that represent l2 and dropout respectively. Two typical weight vectors were used in M2.1, W1 = [0.3, 0.5, 1, 1, 1, 0.7, 0.3]', and W2 = [0.5, 1, 1, 1, 1, 0.5]'. W1 approximately depicts the true value of the relative membership of each type of agencies in the whole data set. W2 gives a more arbitrary weight. Both of them intentionally reduce the loss corresponding to types with smaller populations. The loss of M2.2 used the penalty matrix introduced in the method part before.

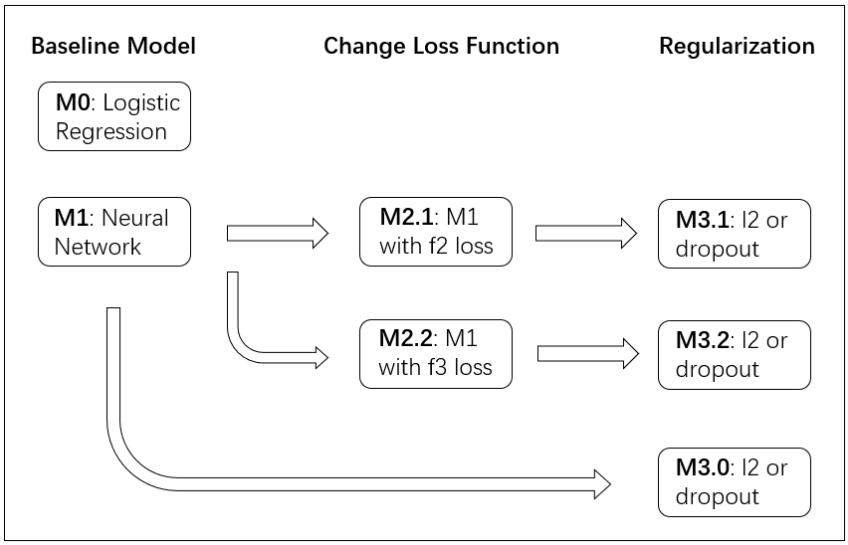

Figure 2: Pipeline plot of the logistics

	M0	**M1**	**M2.1-W1**	**M2.1-W2**	**M2.2**
Accuracy	0.687	0.689	0.702	0.734	0.729
		M3.0.1	**M3.1.1-W1**	**M3.1.1-W2**	**M3.2.1**
Optimal λ		0.016	0.012	0.012	0.016
Accuracy		0.708	0.719	0.705	0.732
		M3.0.2	**M3.1.2-W1**	**M3.1.2-W2**	**M3.2.2**
Optimal p		0.06	0.18	0.12	0.22
Accuracy		0.713	0.714	0.746	0.744

Table 2: Results for different models

From the table above, all the neural network models produce higher prediction accuracy than the crude multinomial logistic regression. The two models with tentative loss functions did a slightly better job than model M1 with the plain cross-entropy loss. M2.1-W2 seems to give higher prediction accuracy than M2.1-W1, which means the assignment of the weight vectors needs more investigation. Regularization implemented by both l2 and dropout methods can

enhance prediction accuracy. The optimized λ in l2 regularization and the optimized dropout probability are showed within each model. As mentioned before, the optimized parameters were obtained from a series of 5-fold cross-validation within the training data set. Hence, the chosen λ and p were corresponding to the best performance within the training data rather than the test data. The figure below shows the sample plots that trace the averaged prediction accuracy given different values λ and p within models M3.0.1 and M3.0.2. It is clear that the optimal parameters do not help to raise the accuracy significantly. Another key observation is that the plots did not show smooth curves as we observed in the linear regression exercise before. One of the possible interpretations is that the randomness of the neural network model makes the prediction accuracy unstable and hence has a random error. More about this would be discussed in the next part. It seems that the optimal dropout probability is not so stable as the λ parameter across different models.

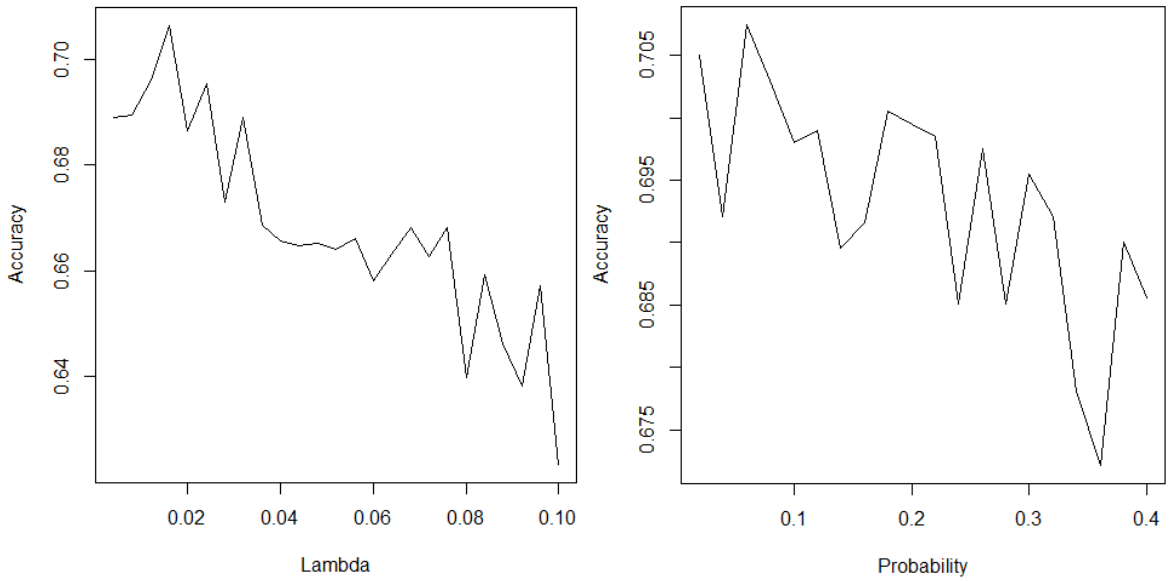

Figure 3: Sample trace plot of λ and p for model M3.0

Because the categorical star rating has 7 levels, I tried to use tables to visualize the predicted type as well as the true type in 7x7 tables. Each cell of the table specifies the number of observations with true type i and predicted type j where i and j have the domain {less or equal to 2, 2.5, 3, 3.5, 4, 4.5, 5}. Hence, the sum of the numbers in the diagonal cells equal to the total number of correctly classified cases. This kind of visual aids also helps to understand the pattern of misspecification in different models. I picked some outputs with interesting and interpretable

patterns. The two tables below compare the prediction vs. true visualization of M3.0.2 and M3.2.2. As mentioned before, M3.2.2 used the loss with the intuition that more severe misspecification on aspect of the magnitude of the star-rating would be penalized with larger loss. Comparing the tables below, it seems that M3.2.2 effectively "forced" the misspecified prediction outcomes to fall into the classes which are the nearest neighborhoods of the true class. However, since the prediction accuracy of M3.0.2 and M3.2.2 did not differ tremendously, the overall pattern of misspecification pattern in the two models are similar.

	Pred. type 1	Pred. type 2	Pred. type 3	Pred. type 4	Pred. type 5	Pred. type 6	Pred. type 7
True type 1	78	25	2	0	0	0	0
True type 2	26	179	71	0	0	0	0
True type 3	1	24	243	52	1	0	0
True type 4	0	0	68	309	55	0	0
True type 5	0	0	0	64	302	53	0
True type 6	0	0	0	2	43	196	41
True type 7	0	0	0	0	1	48	125

Table 4: Prediction vs. true visualization in M3.0.2

	Pred. type 1	Pred. type 2	Pred. type 3	Pred. type 4	Pred. type 5	Pred. type 6	Pred. type 7
True type 1	84	20	0	0	0	0	0
True type 2	34	187	79	0	0	0	0
True type 3	0	23	248	48	0	0	0
True type 4	0	0	57	332	59	0	0
True type 5	0	0	0	59	294	28	0
True type 6	0	0	0	0	47	227	18
True type 7	0	0	0	0	0	41	124

Table 5: Prediction vs. true visualization in M3.2.2

Discussion

Based on the results, the overall performance of neural network models is better than the naively specified multinomial logistic regression. Since insufficient exploration was done within the realm of logistic regression (no complicated polynomial terms, no hierarchical model concern, no regularization, etc.), it is hard to conclude that deep neural networks should be the better way to handle this kind of classification problem. Since it is my first time to implement real data analysis and innovative attempts using neural network, the main purpose of showing the results in a logistics regression setting is to set a bottom line for the outcomes obtained from

the neural network approach, and prevent me starting from an invalid point.

The highlight of this project is the comparison between models under the plain categorical cross-entropy loss, and the two innovative loss functions that incorporate information given by the story behind the data set or our intuition. As mentioned before, when comparing the prediction accuracy of the weighted cross-entropy loss models (M2.1-W1, M2.2-W2), the outcomes show that it is hard to conclude a rule of thumbs for the weight assignment. In future works, more detailed grid search as well as theoretical derivations are all needed to optimize the weights. Among the result, no prediction with perfect prediction accuracy was obtained. Although the proposed methods improve the prediction a little bit, it is clear that trained models are prone to misspecifications between similar classes. This may reveal the question that if accurate classification can be achieved under this setting by the nature of the data set? It is possible that the true hidden mechanism of classifying a home health care agency based on several observed profile information can have a large random error. For example, the same home health care agency can be randomly rated as a 3-star or a 3.5-star agency. To better understand this problem, more detail about the data and the rating program is needed.

This project has some limitations. Firstly, as mentioned before, the randomness in each neural network model was observed but was not adequately reported. Setting a consistent seed seems not to solve the problem because how different models utilize the initial values is invisible. After realizing the unstable outcomes, since the variability was very small, even negligible for most of the cases, I did not give an entire report of the standard error of the outcome over for example 1000 time runs. Without verifying a universally negligible random error, the validity of the comparison conducted might be skeptical, especially when the difference was small. Hence, the λ-accuracy and p-accuracy plot should better be refined too through the repetitive process. Secondly, for the structure of the neural network, the grid search process did not vary the number of nodes in different layers for simplicity. In future work, the search process should be more carefully planned and implemented. Thirdly, only limited regularization methods were tried and no combined methods were considered. Fourthly, the assignments of the penalty matrix in f_3 was a little arbitrary. More theoretical derivation should be supplemented. Fifth, due to the limited sample size (complete-case analysis), covariates like “state” were removed from the covariates lists, which was a loss of information

and potentially induced bias outcomes.

The overall analysis managed to solve the research problem and gave some insights about the data set used. The best model trained can control the misspecification within the nearest neighborhoods of the correct class. In future work, by incorporating more official information about the star-rating definition and more related covariates, better models could be obtained.

Reference

[1] Astone, N et al., (2020). Children and Youth in An Aging America. *Urban Institute*. https://www.urban.org/research/publication/children-and-youth-aging-america

[2] Geng, F., Mansouri, S., Stevenson, D. G., & Grabowski, D. C. (2020). Evolution of the home health care market: The expansion and quality performance of multi-agency chains. *Health Services Research*, *55*, 1073-1084.

[3] Home Health Care – Patient Survey (HHCAHPS), (2020). The Centers for Medicare & Medicaid Services. Version 4.0. https://data.cms.gov/provider-data/dataset/6jpm-sxkc

[4] Ho, Y., & Wookey, S. (2019). The Real-World-Weight Cross-Entropy Loss Function: Modeling the Costs of Mislabeling. *IEEE Access*, *8*, 4806-4813.

[5] Vo, K., Pham, D., Nguyen, M., Mai, T., & Quan, T. (2017). Combination of domain knowledge and deep learning for sentiment analysis. In *International Workshop on Multi-disciplinary Trends in Artificial Intelligence* (pp. 162-173). Springer, Cham.

[6] Chollet, F., & Allaire, J. J. (2018). *Deep Learning mit R und Keras: Das Praxis-Handbuch von den Entwicklern von Keras und RStudio*. MITP-Verlags GmbH & Co. KG.

Appendix

List of the included covariates with comments.

Type	**Variable Description**
Categorical	Type of Ownership
Binary	Offers Occupational Therapy Services
Binary	Offers Speech Pathology Services
Binary	Offers Medical Social Services

Binary	Offers Home Health Aide Services
Continuous	How often the home health team began their patients' care in a timely manner
Continuous	How often the home health team taught patients (or their family caregivers) about their drugs
Continuous	How often the home health team checked patients' risk of falling
Continuous	How often the home health team checked patients for depression
Continuous	How often the home health team determined whether patients received a flu shot for the current flu season
Continuous	How often the home health team made sure that their patients received a pneumococcal vaccine (pneumonia shot)
Continuous	With diabetes, how often the home health team got doctor's orders, gave foot care, and taught patients about foot care
Continuous	How often patients got better at walking or moving around
Continuous	How often patients got better at getting in and out of bed
Continuous	How often patients got better at bathing
Continuous	How often patients' breathing improved
Continuous	How often patients' wounds improved or healed after an operation
Continuous	How often patients got better at taking their drugs correctly by mouth
Continuous	How often home health patients had to be admitted to the hospital
Continuous	How often patients receiving home health care needed urgent, unplanned care in the ER without being admitted
Continuous	Changes in skin integrity post-acute care: pressure ulcer/injury
Continuous	How often physician-recommended actions to address medication issues were completely timely
Continuous	DTC Numerator
Continuous	DTC Observed Rate
Continuous	DTC Risk-Standardized Rate
Categorical	DTC Performance Categorization

Continuous	PPR Numerator
Continuous	PPR Observed Rate
Continuous	PPR Risk-Standardized Rate
Categorical	PPR Performance Categorization
Continuous	How much Medicare spends on an episode of care at this agency, compared to Medicare spending across all agencies nationally
Continuous	No. of episodes to calc how much Medicare spends per episode of care at agency, compared to spending at all agencies (national)