Predict the Number of Days a Patient Will Be Hospitalized in the Next Year

Yutao Zhou, Kaifan Ouyang, Jingjun Hao

Introduction

Predicting the number of days a patient will be hospitalized in the next year is a very important task in the medical field. The accurate prediction can help hospitals manage the resources efficiently and reduce unnecessary hospitalization. The problem we focus on is to estimate the total number of hospital stays in the next year based on the patient's historical medical data by building different prediction models.We use Heritage Health Prize dataset, which includes plenty of medical records, such as MemberID, provideID,Speciality and so on.These data provide the necessary foundation for building predictive models that can capture the complex relationship between health status and need for hospitalization. In this study, we will explore and compare different models, including neural networks (NN), random forest regression(RFR), and logistic regression (LR), and determine the most effective technique to solve this prediction task.

Literature review

For the Heritage Health Prize, the dataset is medical data, which is complex and high-dimensional and contains multiple characteristics. Random forest (RF), as an ensemble learning method that makes predictions by building multiple decision trees, is ideal for working with datasets with a large number of features, As in "Stability of Random Forests and Coverage of random-Forest Prediction Intervals", which explains the stability of Random forests in predictions, The theoretical and empirical performance of the prediction interval coverage is discussed, especially for the continuous response variables [2]. The goal of the Heritage Health Prize is to predict the number of days each patient will stay in the hospital over the next year, which involves complex factors that can have complex relationships with each other. Under the condition that the response variables have finite variance, random forest has high prediction stability, which can help reduce the uncertainty caused by noise, and thus improve the accuracy of the prediction of hospital stay. Random forests reduce the variance of the model by integrating multiple independently trained decision trees, a feature that is particularly important for healthcare data, providing it with reliable predictions [2]. It is mentioned in the paper that random forests can realize the construction of prediction intervals with low computational cost. Forecasting health care costs requires not only accurate projections, but also a clear assessment of the uncertainty and confidence in the projections, reducing unnecessary hospitalizations and reducing health care costs [2].

Model overfitting has always been a problem for all models, especially when dealing with sparse features. Therefore, we refer to the paper *"Learning sparse features can lead to overfitting in neural networks"*. This paper focuses on the role of sparse features in neural networks and how they lead to overfitting problems [1]. In order to solve the overfitting problem caused by sparse features, researchers have proposed several technical strategies. First, by adding regularization terms (such as L1 or L2 regularization), the model's excessive reliance on sparse features can be suppressed [1]. Later in the neural work model, we will also use the alpha parameter to improve the generalization ability of the model. Second, through principal component analysis (PCA) or other dimensionality reduction techniques, redundant information in the data can be removed, and only the features that are meaningful to the task can be retained, thereby reducing the complexity of the model. In data preprocessing, we will also use similar methods for feature selection. In addition, the author also introduced the use of sparse autoencoders to convert original sparse features into compressed dense features, thereby reducing the model's reliance on sparse features [1].

Methods

Data Preprocessing: We processed 13 datasets from different departments, including Claims.csv, DrugCount.csv, LabCount.csv, and Members.csv, by converting categorical columns into numerical values. We grouped the claims data by Year and MemberID, performing aggregations like counting claims. We constructed new binary features by counting occurrences of specific conditions, specialties, and procedures. Similarly, we processed the drug and lab data by aggregating DrugCount and LabCount over Year and MemberID, and saved the results into separate files. Additionally, we engineered demographic features by one-hot encoding the AgeAtFirstClaim and Sex columns to create age group and gender indicators. After processing the individual datasets, we merged them into a single dataset using outer joins on MemberID and Year. We then filled any remaining missing values in the merged dataset with 0, merging it with target hospital stay data.

Feature Construction: Then we focus on extracting and analyzing features to predict the number of days a patient spent in the hospital during a specific year, TARGET. We first filled some nulls with the mean, then grouped and visualized various features such as Specialty, ProcedureGroup, LabCount, and DrugCount to understand their distributions.. We engineered

health-related variables, ranked feature importance. Finally, we performed correlation analysis between the features and TARGET, resulting in a comprehensive feature set which have the following attributes: MemberID, DaysInHospital_Y2, DaysInHospital_Y3. ClaimedTruncated: Members with truncated claims.

Logistics: I choose logistic regression due to its ability to handle multiclass classification, provide probabilistic predictions, support regularization, and avoid the unbounded predictions typical of linear regression. First, I loaded the dataset containing patient records, which included various attributes such as lab counts, drug counts, the number of claims, and demographic details like age and sex. After inspecting the dataset, I removed irrelevant columns such as MemberID and Year. The target variable, representing the hospital stay duration (TARGET), was extracted, and the remaining columns served as input features. To prepare the data for modeling, I applied standard scaling to normalize the features, ensuring that all attributes had a similar scale. I then split the data into training and test sets, reserving 30% for testing. Given the imbalanced nature of the dataset, I computed class weights to handle this imbalance.

Next, a grid search (GridSearchCV) with 20-fold cross-validation was used to optimize the model's hyperparameters, including regularization strength (A smaller value indicates stronger regularization), L1 (Forces some coefficients to be exactly zero to reduce the number of features used) or L2 penalties (Shrinks coefficients but does not set any to zero to prevent overfitting), and the saga solver (efficient for large datasets). After tuning, the logistic regression model was trained on the X_train and y_train datasets using the best combination of hyperparameters. Finally, a confusion matrix was visualized to further analyze the model's predictive performance across the different classes of hospital stays. In addition, I plotted the macro-average ROC curve to visualize the model's ability to distinguish between different classes.

Neural Network: Since the dataset contains many different features (such as LabCount_total, DrugCount_total, etc.), each feature may have a complex nonlinear relationship with the target value TARGET. The neural network can learn these complex relationships well due to its multi-layer structure and nonlinear activation function.We use the MLPRegressor model to solve the neural network implementation of the regression problem. The model can learn complex nonlinear relationships between input features and target values. For parameter setting, the ReLU activation function is used, which is a common nonlinear activation function in modern neural networks. It can handle nonlinear data and speed up convergence. A 2-layer hidden layer with 10 neurons is set to cope with complex data sets. The learning rate is set to adaptive to ensure that the step size is gradually reduced in the later stages of the training process to achieve better convergence. The optimizer is selected as 'adam', combining elements of momentum and adaptive learning rates, which is the current technique of choice for deep learning. Tol is set to 0.001. This parameter determines that during the training process, if the reduction in the model's loss function is less than the tol value, the model is considered to have converged and no further iterations are performed. In the model evaluation, since it is a regression problem, we use the root mean square error (RMSE), RMSLE, R^2 and MAE, which are commonly used evaluation indicators to help evaluate the model performance. Alpha is set to 0.0001 to make the model more flexible. For parameter adjustment, we adjust the number of neurons and increase or decrease the number of layers to adapt to the complexity of the dataset. And we also adjust alpha to control the strength of the L2 regularization term to prevent the model from overfitting or underfitting. 20-fold cross validation outputs the mean and variance of the MSE, providing a reliable assessment of model performance, avoiding model overfitting or underfitting, while maximizing the efficiency of data use.

Random-forest-regression: data_Y1 and data_Y2 are used, feature memberID and Year are removed, target is separated as target variable y, and the remaining columns are used as feature value x. In order to enhance the performance of the model, input features are standardized to ensure that different features are in the same numerical scale. To improve the learning effect of the model. The prediction effect of the model is improved by constructing multiple decision trees and averaging their prediction results.

In the random forest model tuning process, we explore combinations of multiple hyperparameters to ensure that the model does not fit while maintaining low errors:

1. Number of trees (n_estimators) : Gradually increase the number of trees to determine the optimal balance point (25 trees).
2. max_depth: Prevents overgrowth of the tree by limiting the depth (10 layers) and improves the generalization ability of the model.
3. Regularization (min_samples_split, min_samples_leaf) : Avoids overfitting of noise by increasing the minimum number of samples per split and per leaf node.

Also through 20-fold cross validation, subsets the data set and repeatedly train and test the model to ensure that every data

point is used for validation, thereby avoiding overfitting and underfitting problems. Regression uses all RMSE, R^2, MAE, and MSE to help evaluate model performance

Results and Discussion

logistic regression:

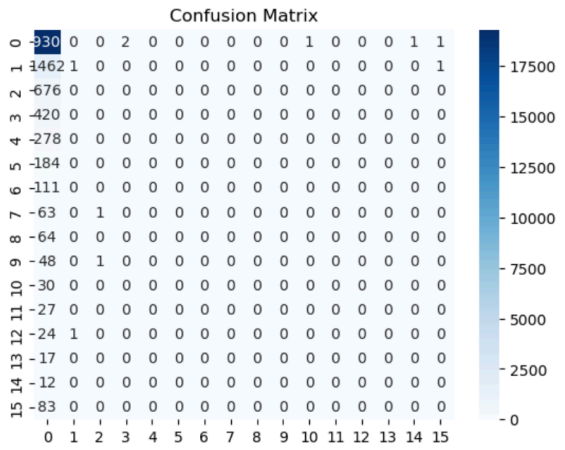

Figure 1. Confusion Matrix

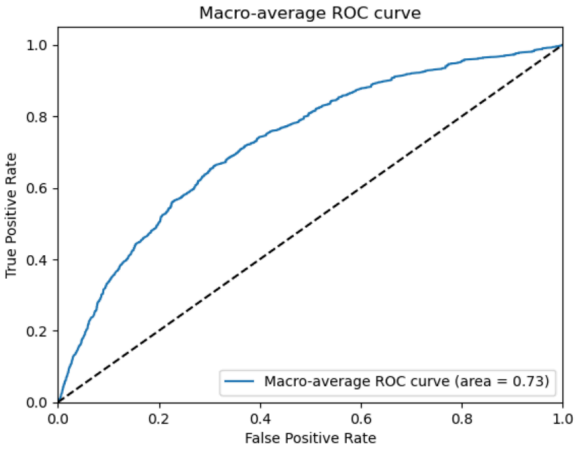

Figure 2. ROC

Accuracy	Recall (weighted)	F1 Score (weighted)	Log-Loss
0.8493	0.8493	0.7802	0.6666

Table 1: Various indicators

Logistic regression assumes a linear relationship between input features and outcomes, which fails to capture the complex, non-linear interactions present in large healthcare datasets. The results in figure 1 and figure 2 also confirmed this: The ROC curve with an AUC of 0.73 suggests moderate discrimination between classes, but the high false positive rate leads to many incorrect predictions for patients with hospital stays. In addition, the confusion matrix shows a strong bias towards predicting the non-hospitalized class. It overwhelmingly predicts patients with no hospital stays while almost entirely failing to predict any of the minority classes. It is concerning in healthcare, where patients who need hospitalization are more important. Therefore, the high accuracy and weighted recall in table 1, which are driven by the model's ability to correctly predict the majority class, do not imply that the model is effective overall.

neural network:

	Before hyperparameter tuning		After hyperparameter tuning	
Metric	Training Result	Testing Result	Training Result	Testing Result
RMSLE	0.51	0.57	0.53	0.54
R2	0.21	-0.19	0.07	0.04
RMSE	1.44	1.75	1.56	1.57
MAE	0.586762109	0.692968613	0.650321272	0.667806418

Table 2. Evaluation results of the training and testing sets

mean MSE score	2.444348
variance	0.046888

Table 3. 20-fold cross validation result

Table 2 shows the evaluation results of the training and testing sets before and after hyperparameter adjustment. Prior to tuning, the model performed well on the training set but poorly on the testing set, indicating overfitting. To address this, we can reduce the model's complexity. Specifically, we increased the alpha value and decreased the number of neurons or layers in the hidden layers. The hidden layer structure was adjusted from (10, 10) to 7, and the alpha value was increased from 0.0001 to 0.0002. After tuning, although the model's performance on the training set slightly declined, the performance on the testing set improved significantly. For instance, the R^2 value increased from -0.19 to 0.04, and the RMSE decreased from 1.75 to 1.57. These improvements indicate that the model has reduced overfitting and generalizes better to unseen data.

Table 3 shows 20 fold cross validation result for neutral network, we can see that the mean MSE is relatively large, indicating that the performance of the model is not very good, but the variance is 0.0468, indicating that the MSE difference between different folds is small, which means that the model is more stable.

random forest regression:

	Before hyperparameter tuning		After hyperparameter tuning	
Metric	Training Result	Testing Result	Training Result	Testing Result
MSE	0.357686	2.415164	2.271384	2.277183
R2	0.864867	0.029034	0.141877	0.084507
RMSLE	0.221361	0.534088	0.488013	0.494058
MAE	0.278939	0.727330	0.703105	0.694625

Table 4. Evaluation results of default & optimize model

Cross-Validation Evaluation Results:

	Metric	Mean	Variance
0	MSE	2.385754	0.066221
1	MAE	0.708667	0.000604
2	R^2	0.081078	0.000244
3	RMSLE	0.499877	0.000070

Table 5. 20-fold Cross Validation Evaluation Result

In Table 4, the default model has an MSE of 0.3577 in the training set, an increase to 2.4152 in the test set, and an increase in MAE from 0.2789 to 0.7273, showing fitting problems. The MSE of the optimized model in the training set and the test set are 2.2714 and 2.2772, and the MAE is 0.7031 and 0.6946, respectively. The errors of the training set and the test set are close, and the model is more stable. The default model training set R^2 is 0.8649, but the test set drops sharply to 0.0290, with severe overfitting. The optimized model training set R^2 is 0.1419, the test set is increased to 0.0845, and the generalization ability is improved. The default model test set RMSLE is 0.5341, and the optimized model test set RMSLE is reduced to 0.4941, narrowing the error gap between training and testing, and making the model more robust. After hyperparameter tuning, the model effectively reduces overfitting, the performance of training and test sets is closer, the generalization ability is significantly improved, and the stability is better.

As Table 5 shows 20-fold Cross Validation, through the same indicator to evaluate the model performance: The variance of the model in cross-validation is small, indicating that the model has good stability and the prediction results are consistent. However, the low R^2 value indicates that the model has limited ability to explain the variance and may still need to be further optimized.

Conclusion

In this study, we explore the logistic regression(LR),neural network(NN) and random forest regression(RFR) in predicting the number of days a patient will be hospitalized in the next year using the heritage health prize dataset.According to the results, the neural network can handle the nonlinear relationship in the patient data, and performs better when faced with a large amount of data. However, when there are a large number of sparse features, it is easier to overfit.While logistic regression provides simplicity and interpretability, it consumes significant time and resources and is hard to handle class imbalance. Random forest can handle complex feature interactions and relationships in data, but it is less interpretive, not enough support in medical scenarios. Compared with LR and RFR, NN is better at handling nonlinear relationships. Although it may be overfitting, it can still produce good results after sufficient hyperparameter tuning and regularization.Thus,so far, we think the neural network is the most effective technique to solve this prediction task.

References

[1] Petrini, Leonardo & Cagnetta, Francesco & Vanden-Eijnden, Eric & Wyart, Matthieu. (2023). Learning sparse features can lead to overfitting in neural networks *. Journal of Statistical Mechanics: Theory and Experiment. 2023. 114003. 10.1088/1742-5468/ad01b9.

[2] Wang, Y., Wu, H., Nettleton, D. (2023). Stability of Random Forests and Coverage of Random-Forest Prediction Intervals. In: NeurIPS 2023 Conference Proceedings. Neural Information Processing Systems Foundation. https://doi.org/10.48550/arXiv.2301.12600