Problem Set 2: Classification for Treatment Using Logistic Regression

Jianing (Julia) Chen

MS Applied Data Science, University of Southern California,

Los Angeles, California 90089, USA

(Dated: February 18, 2021)

Abstract

This report works on the historical medical data with 11 features and utilizes hypothesis testing, sampling techniques to classify if a particular treatment should prescribe for the patients or not. The model builds under logistic regression and tunes parameters with grid search cross-validation. To successfully classify treatment recommendations will be useful for the hospital making decisions in collecting patients' information. Considering that the time consuming and the difficulty of collection for some features, the final model only integrate 5 features: age, blood pressure, gender, blood test, and TestB.

I. INTRODUCTION

This project aims to build a logistic regression model that could classify if a particular treatment is commended for the patient or not. For the variable selection, I mainly perform hypothesis testing and correlation coefficients. For the model tuning, I apply grid search cross-validation. Since the response variable is imbalanced, I check the situation on both oversampling and undersampling. The final model will evaluate how well it classifies the recommended treatment through ROC AUC score.

II. DATA EXPLORATION

The whole dataset contains 10000 instances and 11 variables. There are 7 numeric features and 4 categorical features:

- Categorical features and detailed levels: gender (non-female, female), blood test (negative, positive), family history (False, True, nan[1]) and GeneA (double, none, single).
- Numeric features and summary of statistics shown in Fig. 1:

	treatment	age	blood_pressure	MeasureA	TestB	GeneB	GeneC
count	10000.000000	10000.000000	10000.000000	10000.000000	10000.000000	10000.000000	10000.000000
mean	0.358700	60.032800	84.642355	-5.992414	0.004466	0.547900	0.550400
std	0.479643	8.106546	5.266045	4.168837	0.322338	0.497725	0.497478
min	0.000000	29.000000	-99.000000	-21.708000	-0.564197	0.000000	0.000000
25%	0.000000	55.000000	82.314030	-8.790691	-0.251337	0.000000	0.000000
50%	0.000000	60.000000	83.997305	-5.956422	-0.039662	1.000000	1.000000
75%	1.000000	66.000000	86.342954	-3.217110	0.215928	1.000000	1.000000
max	1.000000	92.000000	107.595583	8.889658	1.231447	1.000000	1.000000

FIG. 1. Summary of Numeric Features

We can see that variable treatment, GeneB, GeneC are binary variables, while age, blood pressure, MeasureA and TestB are continues variables. We could also observe that: age has mean (50%) close to median; blood pressure has negavtive values; MeasureA has mean close to median; the mean for TestB is higher than the median.

The histogram for each continues variables shown in Fig.2 shows that the range for blood pressure is very small with very large values; age and MeasureA are seemingly

[1] nan: missing variables

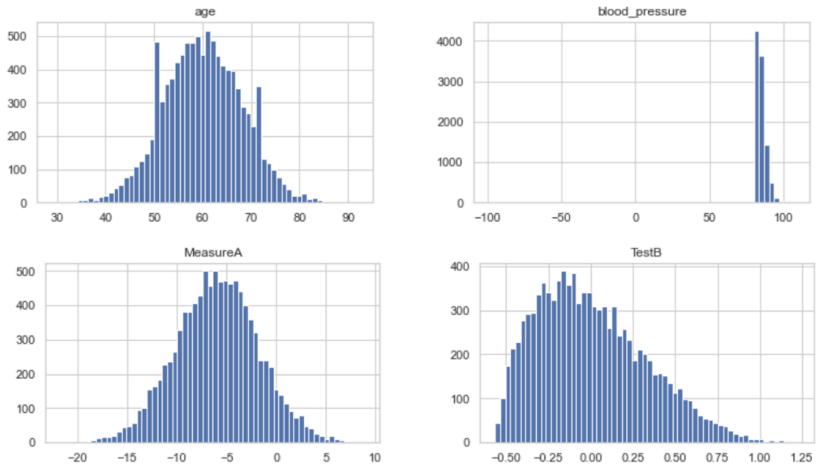

FIG. 2. Histogram for Continues Variables

normal distributed; TestB is bit right skewed. We will adjust them in Data Preprocessing Section.

Next, I check the missing values for each features. I find there are 2932 missing variable in family history variable. And this variable is heavy imbalanced, because it has 6968 false responses but only have 100 true responses. It will be important to manipulate this before we building model. Furthermore, I also realize that the response variable, treatment, is also imbalanced, so I plan to weight the classes by its representation Model Selection.

III. DATA PREPROCESSING

The first step is to fill in the missing variables from family history. Since there are 2932 missing values, it will be inappropriate to remove the missing variables. I combine the true response and the missing response as other responses and keep the false responses unchanged to prevent heavy imbalance. That gives me 6968 false responses and 3032 other responses.

The Second step is to encode the categorical variables. Since gender and blood test are binary variables, we can directly apply the ordinal encoding. While GeneA and family history are imbalanced, it prevents me from getting an ANOVA table due to convergence difficulty when I use one-hot encoding. So I ended up using ordinal encoding for all categorical variables. The results of ordinal encoding shows on TABLE I.

Next, to perform the model evaluation, I split the dataset into three parts: 60% training

TABLE I. Ordinal Encoding Result

Variable	Encoded
gender	0 female, 1 non-female
blood test	0 negative, 1 positive
family history	0 false response, 1 other response
GeneA	0 double, 1 none, 2 single

set, 20% validation set, and 20% testing set. I will train my model based on the training set and use the validation set to tune my model. The final result will test on the testing set.

I operate exploratory data analysis on training set and below are some data cleaning and manipulations:

- The minimum blood pressure is -99, which does not have a practical meaning. So I will consider -99 as a mistake in inputting. In total, including the validation and testing set, there are five rows of blood pressure -99, so I will remove those rows in all datasets. In addition, the range of blood pressure is very small, but the number is quiet large, so I used log-transformation on blood pressure, refer Fig.3.

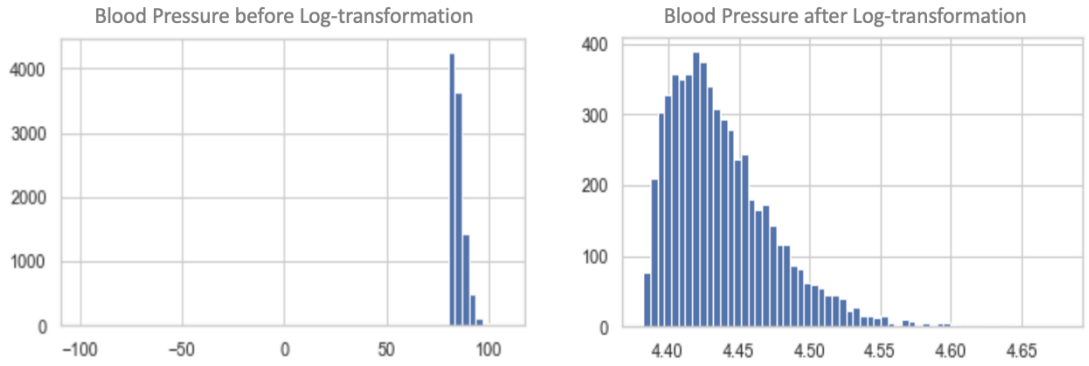

FIG. 3. Log-transformation Result

- From Fig. 1 we see that MeasureA is normal distributed, when I plot the boxplot for MeasureA, I identify many outliers. Therefore, I decide to apply winsorization [2].

There is also an interesting finding, Fig. 4 (left), such that the distribution of treatment has two peaks, so this is a bimodal distribution. In contrast, no treatment group is still a

[2] winsorization: It is supposed to be used in symmetric distribution, which means if we are replacing the n largest values, we also need to replace the smallest n values.

normal distribution. As the distributions are clearly different for the TestB of treatment vs. no treatment, this variable would likely be a significant predictor in our final model. Indeed, I try to amplify this difference by taking the absolute value for the testB on Fig. 4 (right) the new column of data after taking the absolute value is called newTestB, in the report we will use the TestB instead of newTestB in the next few sections.

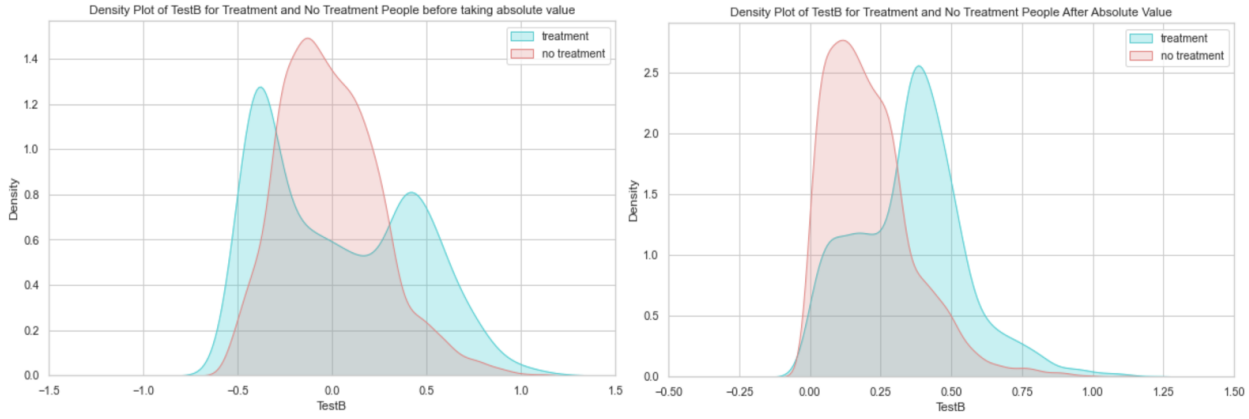

FIG. 4. Density Plot of TestB

IV. FEATURE SELECTION

From the correlation matrix, I find that gender and MeasureA have relatively strong positive correlations. The age has a relatively strong negative correlation. Besides, considering the correlation between age and MeasureA are very high, which is -0.97. This is what we called multicollinearity, and it can be an issue and reduce the fitted model's performance. To avoid multicollinearity, I will only include either age or MeasureA in my model. The detailed absolute correlation coefficient concerning price shows on TABLE II.

To test the importance for each variables, I fit all variables into a logistic regression and perform a hypothesis testing. The null hypothesis is there is no relationship between treatment and a specific variable. We will reject the null hypothesis in the case that p-value is larger than the significance level.[3] The p-value for each variable is illustrated on Fig.5 under $P > |z|$ column.

Suppose our significant level is 5%, we can see that MeasureA has p-value at 0.31 and

[3] Significance level: the probability of rejecting the null hypothesis, usually will be set at or below 5%.

TABLE II. Absolute Correlation Coefficient

Variable	R^2
TestB	0.368199
gender	0.337044
age	0.155299
MeasureA	0.150385
blood pressure	0.091264
blood test	0.053250
family history	0.046707
GeneC	0.042166
GeneB	0.021847
GeneA	0.003361

	coef	std err	z	P>|z|	[0.025	0.975]
age	-0.0671	0.015	-4.568	0.000	-0.096	-0.038
blood_pressure	0.5299	0.162	3.279	0.001	0.213	0.847
gender	1.5721	0.060	26.007	0.000	1.454	1.691
blood_test	-0.4813	0.107	-4.496	0.000	-0.691	-0.271
MeasureA	-0.0292	0.029	-1.015	0.310	-0.086	0.027
TestB	0.5263	0.091	5.778	0.000	0.348	0.705
GeneB	0.1645	0.059	2.775	0.006	0.048	0.281
GeneC	0.1992	0.059	3.361	0.001	0.083	0.315
GeneA	0.0071	0.040	0.177	0.860	-0.071	0.086
family_history	0.2723	0.064	4.284	0.000	0.148	0.397

FIG. 5. F-test Result

GeneA has p-value at 0.86. We fail to reject null hypothesis for those two variables, therefore, we can conclude that MeasureA and GeneA have no effect no treatment.

So we will purpose to use: age, blood pressure, gender, blood test, TestB, GeneB, GeneC and family history to build model. Considered that MeasureA, TestB, GeneB and GeneC are really expensive and difficult to gather, I will fit model for two sets:

1. age, blood pressure, gender, blood test, TestB, GeneB, GeneC and family history

2. age, blood pressure, gender, blood test, TestB

The upper sets of variables are derived from F-test, the lower set of variables keeps top 5 variables based on correlation coefficient. I understand that TestB, GeneB and GeneC are hard to collect, so I only keep GeneB, because Fig. 4 shows that it really helpful to distinguish treatment groups.

The ROC AUC score for the first sets of variable is 0.8280, the ROC AUC score for the second sets of variable is 0.8251. So there is not much lose of accuracy after we remove GeneB, GeneC and family history, but will reduce the cost for hospital which will be a good trade off.

V. MODEL SELECTION

In previous model selection part, I do not add any regulation to the model. Since there many combination of penalty and solver, I decide to use grid search cross validation, it used as a evaluating metric for the model performance to select the best hyperparameters. In my result, I see many warning saying that my model failed to converge. After consideration, I end up with using L2 penalty, liblinear solver, maximum number of iteration is 100, and C=10.

As I mentioned earlier that response variable is imbalanced. So I use balance sampling, oversampling and undersampling, the result for each case shows below:

TABLE III. Result for Logistic Models

	Precision	Recall	roc-auc score
Simple	0.6852	0.5400	0.8251
Balanced	0.6139	0.7472	0.8247
Oversample	0.6135	0.7467	0.8247
Undersample	0.6123	0.7495	0.8242

The simple model refers to the second model I proposed in feature selection without regularization and resampling. Compared to the other three models, the simple model has higher precision and ROC-AUC score. There always is a trade between Precision and Recall. In this situation, Recall is more critical because if someone needs treatment, we do not want to say he/she does not need it[1]. Finally, we decide to choose the balanced model with 5 predictors: age, blood pressure, gender, blood test, TestB, because it has higher recall than oversampling and higher ROC-AUC score than undersampling.

VI. MODEL EVALUATION

To perform model evaluation, I first clean my testing set, such as take log transformation of blood pressure and remove outliers from MeasureA, and take the absolute value for TestB.

The confusion matrix for final model shows on TABLE IV. The false negative is 267, and false positive is 387. The Recall is $\frac{509}{224+509} = 0.6944$, and the Precision is $\frac{509}{360+509} = 0.5857$.

TABLE IV. Confusion Matrix

	Predict No-Trt	Predict Trt
Actual No Trt	905	360
Actual Trt	224	509

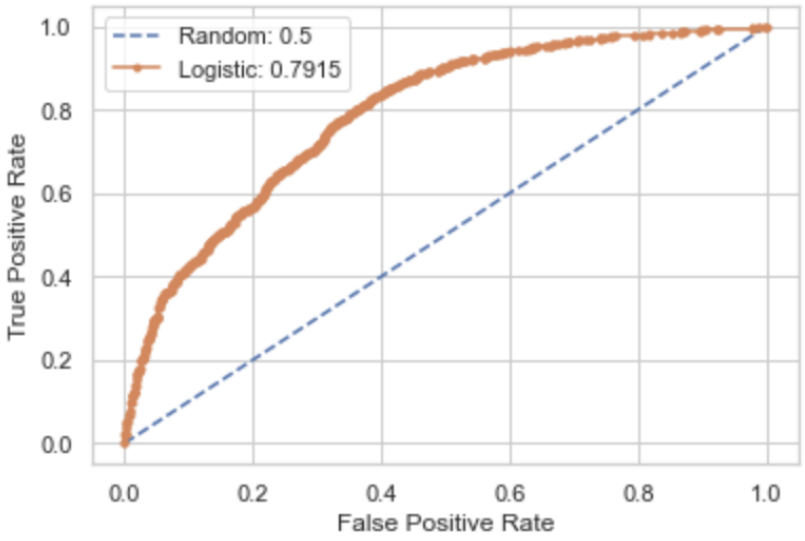

FIG. 6. AUC - ROC Curve

ROC curve is the probability that true positive rate against false positive rate at all possible thresholds, and can summarize the model predictability based on the area under curve (AUC). AUC score is the measures the overall performance of a classifier and ranges from 0 to 1. The higher the AUC score means the better model is[2]. The final model has ROC-AUC score as 0.7915, which is bit lower than the score on validation set (ROC-AUC score is 0.8247), so we would consider that our model is a good fit.

VII. INTERPRETATION

The final model can be expressed as: Y = − 11.7643 − 0.0632 · age + 2.9689 · blood pressure + 1.9442 · gender − 0.6388 · blood test + 5.8323 · TestB, where Y is the natural log of odds for the probability of treatment, Y = $ln\ (\frac{P(treatment)}{1-P(treatment)})$, or the logistic distribution function for $P(treatment)$ is:

$$P(treatment) = \frac{\exp(-11.7643-0.0632*age+2.9689*bloodpressure+1.9442*gender-0.6388*bloodtest+5.8323*TestB)}{1-\exp(-11.7643-0.0632*age+2.9689*bloodpressure+1.9442*gender-0.6388*bloodtest+5.8323*TestB)}$$

TABLE V. Coefficient and Odds Ratio

Variable	coef	Odd Ratio
age	-0.0632	0.9387
blood pressure	2.9689	19.4711
gender	1.9442	6.9884
blood test	-0.6388	0.5279
TestB	5.8323	341.1327

The final result for each coefficient and odd ratio shows on TABLE V. Coefficient indicates the increase or decrease in the log odds, so it can either be positive or negative. Odds ratio compares the odds of two events, if odd ratio is greater than 1 indicate that the even is more likely to occur as the predictor increases. If odds ratio is less than 1 indicate that the event is less likely to occur as the predictor increases. This fitted models says that:

- Holding all other variables at a fix value, the odds ratio of age indicates that every increase the age by 1 unit, the treatment is recommended decrease by 7%.
- Holding all other variables at a fix value, the odds ratio of blood pressure shows that increase the blood pressure by 1 unit, we expect that about 19.5 times increase the odds of patient getting treatment.
- Holding all other variables at a fix value, the odds of getting treatment for non-female (non-female =1) over the odds of getting treatment for female (female =0) is about 7. In terms of percent change, we can say that the odds for non-female are 600% higher than the odds for female.
- Holding all other variables at a fix value, the odds of getting treatment for positive blood test (blood test = 1) over the odds of getting treatment for negative blood test (blood test = 0) decrease by 48%.

- TestB is the most important variable, holding all other variables at a fix value, increase 1 unit of TestB there will be 341 times increase the odds of patient need treatment. (Note: the range of TestB quiet small, so we will not expect to have 1 unit increase)

VIII. CONCLUSIONS

The final model is Y = − 11.7643 − 0.0632 · age + 2.9689 · blood pressure + 1.9442 · gender − 0.6388 · blood test + 5.8323 · Test. The recall is 0.7472 and the precision is 0.6139. The ROC AUC score is 0.8247.

I purpose the model with less variables because I want to be more efficient. To reduce the cost, variables age, blood pressure, gender, blood test, and TestB are good enough to predict treatment recommendations. Among those 5 predictors, TestB and blood pressure are the essential variables. One unit increase in TestB will have 341 times increase in the odds getting treatment, the blood pressure will have 9.5 times increase in the odds getting treatment. Lastly, MeasureA and GeneA do not contribute to the model. GeneB and GeneC are okay but not required.

DATA AVAILABILITY

Data is available at: Github

CODE AVAILABILITY

Code is available at: Github

---

[1] A. Jain, A brief journey on precision and recall, (2018).

[2] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning: With Applications in R* (Springer Publishing Company, Incorporated, 2014).