Journal of Modeling and Optimization 2021;13(2):80-91
https://doi.org/10.32732/jmo.2021.13.2.80

Supervised Learning for Binary Classification on US Adult Income

Li-Pang Chen

Department of Statistics, National Chengchi University, No. 64, Section 2, Zhinan Rd, Wenshan District, Taipei City, 116, Taiwan
E-mail: lchen723@nccu.edu.tw (Corresponding author)

Received: 25 April 2021; Accepted: 30 July 2021; Available online: 10 November 2021

Abstract: In this project, various binary classification methods have been used to make predictions about US adult income level in relation to social factors including age, gender, education, and marital status. We first explore descriptive statistics for the dataset and deal with missing values. After that, we examine some widely used classification methods, including logistic regression, discriminant analysis, support vector machine, random forest, and boosting. Meanwhile, we also provide suitable R functions to demonstrate applications. Various metrics such as ROC curves, accuracy, recall and F-measure are calculated to compare the performance of these models. We find the boosting is the best method in our data analysis due to its highest AUC value and the highest prediction accuracy. In addition, among all predictor variables, we also find three variables that have the largest impact on the US adult income level.
Keywords: Boosting; Categorical data; Income; Discriminant analysis; Logistic regression; Prediction; Random forest; Support Vector Machine; Unbalanced binary classification.

1. Introduction

1.1 Objective

The inequality of wealth and income is a huge concern around the globe, and governments in different countries are using different interventions to address income inequality. In this project, we aim to utilize some of the existing classification methods to help understand these inequality issues. Our strategy is to train a binary classifier, denoted as Y, to predict the whether a person earns more than \$50K or not per year based on the social factors and to find out what factors influence the income level the most.

1.2 Description of Dataset and Challenges of the project

The US Adult income dataset was extracted from the 1994 US Census Database.The data set consists of anonymous social information such as occupation, age, native country, race, capital gain, capital loss, education, work class and others. The data set includes figures on 48,842 different records and 14 attributes for 42 nations. The 14 attributes consist of 8 categorical and 6 continuous attributes containing information, where the detailed information is summarized in Table 1. The dataset can be accessed from UCI repository [1] and Kaggle [2]. The challenges of this dataset were discussed in some of the past literature. First, the dataset contains around 7.4% of missing values, and the handling of these missing data is of great challenge to the prediction outcome. Second, the number of observations is around 48k, which is considered computational expensive for some of the classification algorithms. Third, half of the predictor variables are multi-level categorical, which reduces the amount of information preserved in data. Besides, many similar observations have different response classes, posing a challenge to the accuracy of the prediction. In this project, we address each challenge above to our best knowledge in order to best predict the US adult income level.

1.3 Project-related literature

Supervised learning is perhaps a powerful and useful approach in contemporary machine learning and statistical analysis, whose main purpose is to characterize the relationship of the main interested response and the information of predictors, and then use the predictors to do modeling and prediction. Classification, whose response variable is taken as binary outcome, is one of common applications in the framework of supervised learning.

For the US adult income dataset, it was especially preferred in machine learning models because it is large enough to allow enough room for train and test sets, and to make discerning small differences in performance reliable. In the early approaches, [3] employed Gradient boosting classifier method; [4] made the usage of

Extreme Gradient Boosting (XGBOOST) for prediction tasks; [5] implemented Principal Component Analysis (PCA) to generate and evaluate income prediction data based on the current population survey provided by the U.S. Census Bureau. [6] tried to replicate Bayesian networks, decision tree induction and lazy classifier for the dataset and presented a comparative analysis of the predictive performances. In addition to the existing approaches, there are a lot of machine learning strategies that might be suitable to analyze this dataset, such as discriminant analysis, support vector machine (SVM), random forest, and neural nets [7–9].

1.4 Main Contributions

Different from existing work, in this paper we analyze the US adult income data by examining some popular machine learning methods, including logistic regression, discriminant analysis, support vector machine (SVM), random forest, and boosting. In addition to discussing detailed algorithms, we also provide some commonly used R functions to implement the methods. In addition, based on the boosting method, we further summarize the relative importance and identify most important variables that determine the level of adult income. To the best of knowledge, it was not explored in existing work.

2. Data preparation

2.1 Data cleaning

After browsing the data, we find that the two variables **Education** and **Education.num** are simply two different representations of education level. There is no need to keep both of them when we are training our models. Considering that **Education.num** is a numerical variable and larger value of it means higher education level,we decide to remove **Education** from our dataset and keep **Education.num** only. Moreover, we also find that there are several variables having values '?' that is unreasonable, so we first treat the character '?' as missing value and then check whether it is appropriate to remove all observations with missing values.

By exploring the missing values, we find that about 7.4% of observations have missing values. All the missing values occur in the variables **occupation**, **workclass** and **native.country**, and the proportions of missingness are shown in Figure 1.

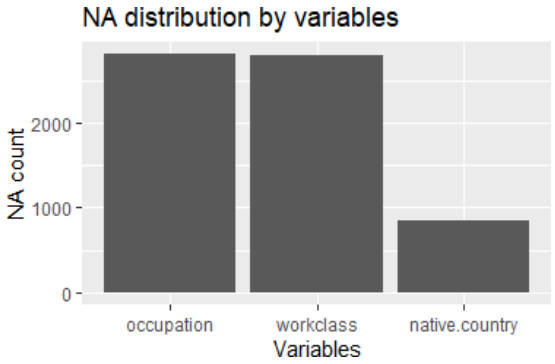


Figure 1. Proportion of missing values

To deal with the missing values, we create a new variable **missing_ind** to indicate if an observation has missing values. It equals *True* when an observation has one or more missing values. Then we explore the relationship between **missing_ind** and the response **salary** as follow:

Figure 2. salary VS missing_ind

From Figure 2, we can see that for those observations with missing values, a larger proportion of them have a low salary level, compared to those observations without missing values. It means that the missing data mechanism here is not missing completely at random (MCAR). Thus, it is inappropriate to remove all the observations with missing values.

Therefore, for this dataset, we are not going to remove the observations with the character ' ?'. Instead, we will treat '?' as a new level 'unknown' and do further analysis using all the observations in the original data.

2.2 Split training set and testing set

In order to avoid overfitting and to measure the performance of our models in a more reasonable way, we randomly split our data into 70% of the training set and 30 % of the testing set. We use the training set to build the models, while the testing set is used to measure the performance of our models.

3. Exploratory data analysis

The training dataset contains 48,800 observations with 15 variables that are denoted as X. A description of the variables X is shown in Table 1. 76.07% of the data is from the low salary group with annual income less than or equal to 50k, while the other 23.93% is from the high salary group with annual income over 50K.

Table 1. Data Description

Variable	Description	Type
Salary	Individual's annual salary	categorical
Workclass	Individual's work category	categorical
Education	Individual's highest education degree	categorical
Marital-status	Individual's marital status	categorical
Occupation	Individual's occupation	categorical
Relationship	Individual's relation in a family	categorical
Race	Race of individual	categorical
Gender	Gender of individual	categorical
Native-country	Individual's native country	categorical
Age	Individual's age	numerical
Fnlwgt	final weight: the weights on the CPS files are controlled to independent estimates of the civilian non-institutional population of the US. These are prepared monthly for us by Population Division here at the Census Bureau.	numerical
Capital-gain	personal capital gain	numerical
Capital-loss	personal capital loss	numerical
Hours-per-week	Individual's working hour per week	numerical
Education-num	Individual's highest education degree, range from 1 to 16. The higher the education-num, the higher the education degree.	numerical

3.1 Categorical variables

Figure 3 shows the distribution of each categorical variable for both salary levels, and it provides us with an initial idea of how different attributes of individuals vary between people from the low salary group and the high salary group. Most of the observations shown in Figure 3 are not out of ordinary. We can see that most people from the high-income group own corporations while most people from the low-income group are working class. The percentage of people with higher **education** (Bachelors and above) is also much higher for the high-income group. The **marriage status** of people from the low-income group seems more "complicated", it seems that money can indeed ruin your marriage sometimes. A higher percentage of people with high incomeare **white-collar** and **gold-collar** workers, while people with low income are more involved in blue-collar jobs. The majority of people with high income are **husbands**, while the relationship of low-income people is more diverse. This is consistent with themarital-status of these two groups. In terms of race, we can see a higher percentage of the **white race with high-income** and a higher percentage of the **black**

race with low-income. This is consistent with the social phenomenon that the white race being the dominant race in middle- and upper-class in the U.S. In terms of gender, it is no surprise that the percentage of **males** is much higher for the high-income group.

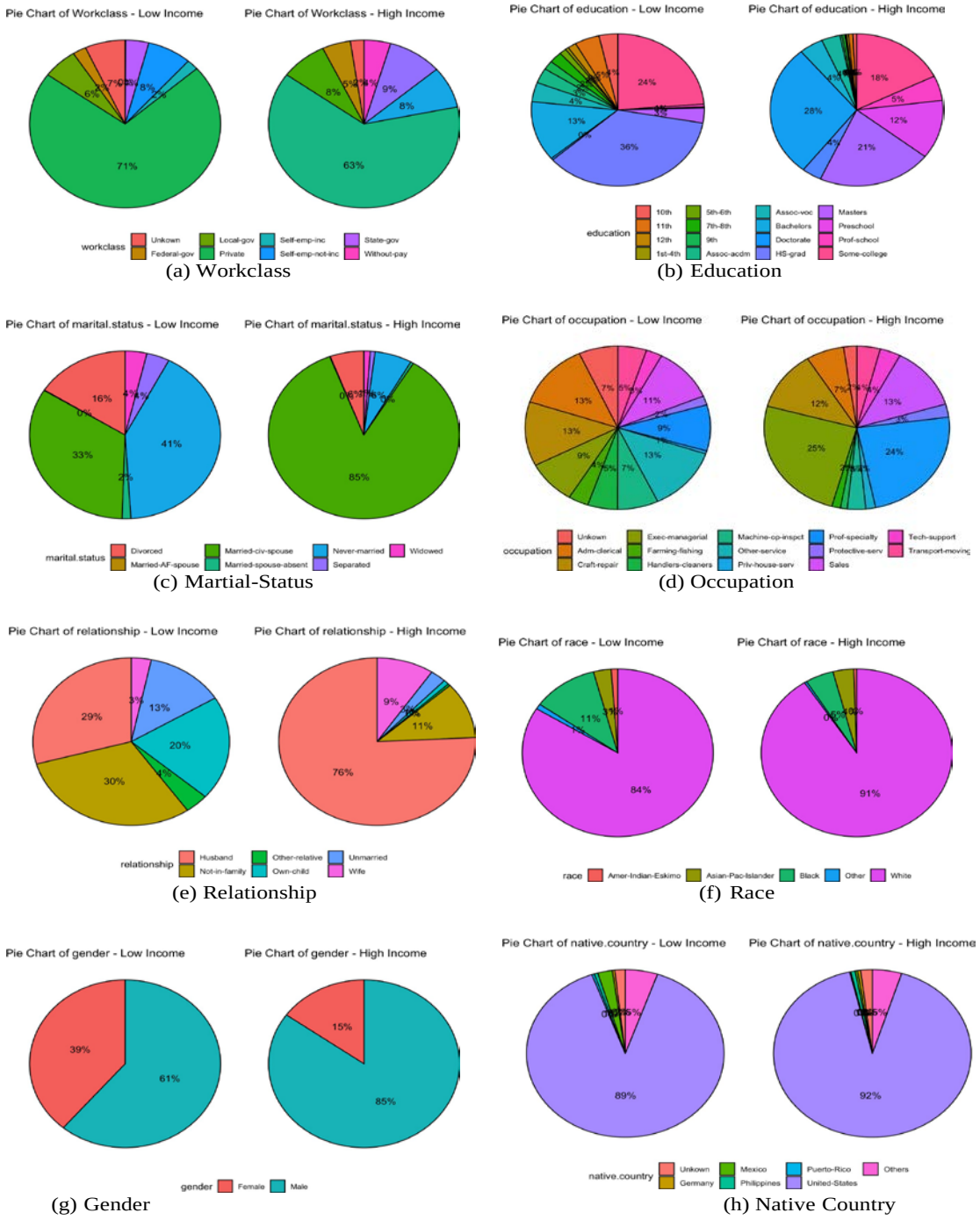


Figure 3. Pie Charts showing the distribution of each categorical variable, for eachsalary level.

3.2 Numerical variables

We explore the relationship between numerical variables to see if there is any significant correlation. As shown in Figure 4, we do not see high correlation between any pairs of variables. In addition, we can also see the distribution of all numerical variables in Figure 5. We realize for most observations, their **capital.gain** and **capital.loss** is 0. Besides,from the boxplots, we can see the distribution of these variables is scattered. Moreover,the histograms show that no continuous variables is normal distribution.

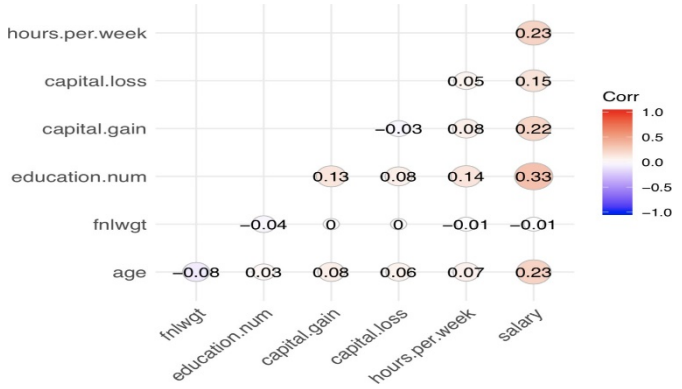
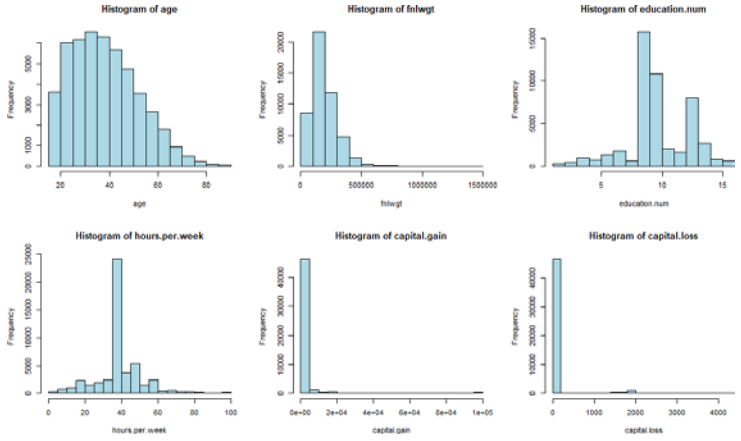
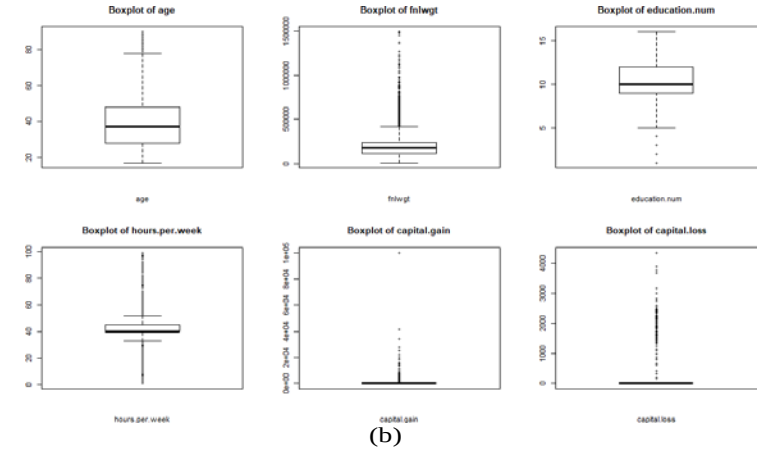


Figure 4. Correlation Plot



(a)



(b)

Figure 5. Distribution of all numerical variables via histogram and boxplot

4. Model selection

4.1 Logistic regression

4.1.1 Model introduction

Logistic regression models the probability of response using the logit function

$$\log\left\{\frac{P(Y)}{1-P(Y)}\right\} = X^T\beta$$

and the parameters β are obtained from the maximum likelihood estimates. As a generalized linear model, logistic regression is a widely used technique because it is highly interpretable and does not require too many computational resources [10]. However, the disadvantage of logistic regression is that it cannot solve non-linear problems and is highly relied on the choice of predictors with good performance [10].

4.1.2 Model Fitting

The income data has the response salary that falls into one of two categories, "<= 50K"or ">50K" that allows us to fit it to a logistic regression model. We use **glm(family= binomial)** function to fit three logistic regression models to compare the results.They are (i) the model with only intercept, (ii) the model only with numeric variables, and (iii) the model with all variables except education.num, respectively. The third model with all the variables except education gives the best performance since all the variables are shown as significant in impacting the salary and it has the lowest AIC and MSE.We used **halfnorm()** to check the outliers of the third model, which is also our final model, and we find that the model fits very well since there are no detected outliersin the model.

4.2 Linear discriminant analysis

4.2.1 Model introduction

Although logistic regression is a simple and powerful linear classification algorithm, it has limitations such as instability for well-separated classes. Linear Discriminant Analysis (LDA) can address this limitation with prior probability obtained for each class from Bayes' theorem. The classification of an observation is done in the following two steps:

(1) Identify the distribution for input X for each of the class ($Y = k$ with $k = 0,1$).

(2) Flip the distribution using Bayes theorem to calculate the probability

$$\Pr(Y = k|X = x) = \frac{\Pr(X=x|Y=k)*\Pr(Y=k)}{\sum_{i=1}^{k}\Pr(X=x|Y=i)*\Pr(Y=i)}$$

where $\Pr(Y = k|X = x)$ represents probability that an observation belongs to responseclass $Y = k$ and $\Pr(X = x|Y = k)$ represents probability of $X = x$, for a particular response class $Y = k$ [11]. The distribution of $X = x$ needs to be calculated from the historical data for every response class $Y = k$. For example, if we want to predict the salary level (>50K or <=50K) of a person based on predictors, we need to first identify the distribution of these characteristics from historical data for the two salary level (>50K or <=50K).Then using Bayes theorem to find the probability of this person belonging to each level (>50K or <=50K) from the distribution of the set of characteristics. The level that gets the highest probability is the output class and a predictionis made. Note that the LDA assumes that predictors are normally distributed and that thedifferent classes have class-specific means and equal variance/covariance.

4.2.2 Model fitting

Before fitting the model, we need to check for the normality and equal-variance assumptions of the LDA model. Apparently, these two assumptions are both violated so it is expected that the LDA may not perform very well. Next, we use the **lda()** function in the **MASS** package to train the model and measure its performance by making predictions on thetest data. The prediction performance of the model is shown in Section 5.1.

4.3 Quadratic discriminant analysis

4.3.1 Model Introduction

Quadratic discriminant analysis (QDA) [11] provides an alternative approach as LDA. Similar to LDA, QDA assumes that the observations from each class are drawn from a Gaussian distribution, and plug estimates for the parameters into Bayes' theorem in order to perform prediction. The difference between QDA and LDA is the homogeneity and heterogeneity of variance between each class, LDA assumes that the covariance matrices are the same for all classes, while QDA assumes the covariance matrices are different. Under this assumption, we have

$$\begin{aligned}\delta_k(x) &= -\frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k) - \frac{1}{2}\log|\Sigma_k| + \log\pi_k \\ &= -\frac{1}{2}x^T\Sigma_k^{-1}x + x^T\Sigma_k^{-1}\mu_k - \frac{1}{2}\mu_k^T\Sigma_k^{-1}\mu_k - \frac{1}{2}\log|\Sigma_k| + \log\pi_k \end{aligned} \quad (1)$$

that assigns an observation $X = x$ to the class for which is largest. By plugging the estimators for Σ_k, μ_k and π_k into (1), we can assign each observation into a class. The quantity x appears to be quadratic term in (1), this is where QDA get its name.

4.3.2 Model fitting

Before using **qda()** function in package **MASS**, we need to filtrate the variables to make sure the data can be fitted into the model. By testing each variable, we find that the variables **workclass**, **education,** and **native.country** will lead to rank deficiency, which may be due to that the factorized data conflicts with the assumption that each variable follows a normal distribution. So we remove those three variables from the original data and fit the QDA model.

Similarly, we make predictions on the test data to measure the performance of the QDA model. We plot the ROC curve and compute different metrics for classificationto measure the performance, as shown in Section 5.1.

4.4 Support vector machine (SVM)

4.4.1 Model introduction

SVM [12] constructs a hyperplane $f(x) = \beta_0 + x^T \beta$ or set of hyperplanes in a high-dimensional space, which can be used for binary and multi-class classification problems. The estimation of parameters of binary classification is obtained from the cost minimization problem, and the class is determined from the decision function $\widehat{G}(x) = sign(\widehat{\beta}0 + x^T \widehat{\beta})$. SVM can also be generalized for classification of non-linear decision boundaries, in which case functions of the predictors in quadratic, cubic or radial terms will be used to address the non-linearity.

4.4.2 Model fitting

The first consideration for fitting the SVM is to choose an appropriate kernel function. Given the size of the training dataset n and the number of covariates q, we think a linear kernel with a running time of $O(n^2q)$ is appropriate for this classification task, since the dataset does not inherit explicit internal structures, and the training time for other kernel functions will be much more computationally expensive especially when the size of the dataset is large.

The function **tune()** is used to perform a ten-fold cross-validation on a set of costparameters ranging from 0.05 to 5. The best model was chosen with an optimal cost parameter C=0.1. The prediction performance of the model is shown in Section 5.1.

4.5 Random forest

4.5.1 Model introduction

Random forest [13] is a popular method in machine learning. It constructs many classification and regression trees (CARTs) [14] randomly, and each CART is independent. When the forest is constructed, it can be used to make predictions based on new inputs. The class of an input is determined by every single CART. Therefore, the predicted class of the inputis the most frequent class generated by CARTs.

A CART is trained by a sample generated from the original sample. Since the random forest consists of many CARTs, bootstrap [15] is applied. Bootstrap generates multiple samples from the original sample with replacement. Each sample has many features. Random forest randomly selects different feature subsets, and then each CART in the forest uses ID3 algorithm [16], C4.5 algorithm [17] or Gini index [14] to select the most important feature from a feature subset to split the tree.

Gini index represents the probability of a randomly selected feature being misclassified. The form is given by

$$Gini(p) = \sum_{k=1}^{K} p_k(1 - p_k) = 1 - \sum_{k=1}^{K} p_k^2$$

with p_k being the probability of a input being classified in a particular class and K being the number of categories in a sample. Then the Gini index of a sample D is

$$Gini(D) = 1 - \sum_{k=1}^{K} \left(\frac{C_K}{N_D}\right)^2$$

where C_K is the number of observations in categorical K, and N_D is the sample size. If a feature A separates the sample D into two samples D_1 and D_2, then

$$Gini(D,A) = \frac{N_{D_1}}{N_D} Gini(D_1) + \frac{N_{D_2}}{N_D} Gini(D_2)$$

A smaller Gini index indicates a better result. Therefore, a CART selects features that reduce maximum Gini index to split from the root to the leaves. In this way, many CARTs are generated to form the random forest.

4.5.2 Model fitting

In the R language, there is a package called **randomForest**, and in this package, a built-in function named **randomForest()** can be used to fit the random forest model. Since random forest consists of many CARTs and randomly selects feature subsets, *mtry* and *ntree* are two important parameters need to be determined. *mtry* indicates size of feature subset selected in each split, and *ntree* demonstrates number of trees trained in the model. First, we fit the data with default *mtry* and set *ntree* = 500. We find the error rate becomes constant after 100 trees. Thus, we can choose any number which is greater than 100. In this model, we set *ntree* = 300. Next, we need to find the best *mtry*. Because this dataset contains 14 predictors, and the variable **education** has been excluded, we try *mtry* from 1 to 12. The results demonstrates the average error reaches the lowest point when *mtry* equals to 3. Finally, we fit the data into the random forest model with *mtry* = 3 and *ntree* = 300, and the AUC of the model is 0.91.

4.6 Boosting

4.6.1. Model introduction

Boosting is a general approach that can be applied to many statistical learning methods for regression or classification [13]. However, we only consider the boosting method to improve the performance of decision trees here.

By using the boosting approach, we grow multiple trees sequentially: each tree is grown using information from previously grown trees. The algorithm of applying boosting for regression trees is shown below [13]:

(1) Set $\hat{f}(x) = 0$ and $r_i = y_i$ for all *i* in the training set.

(2) For $b = 1, \cdots, B$, repeat:

1) Fit a tree $\hat{f}^b$ with d splits to the training data (X, r).

2) Update $\hat{f}$ by adding a shrunken version of the new tree: $\hat{f}(x) := \hat{f}(x) + \lambda \hat{f}^b(x)$

3) Update the residuals with $r_i = r_i - \hat{f}^b(x)$

(3) The final model is

$$\hat{f}(x) = \sum_{b=1}^{B} \lambda \hat{f}^b(x)$$

Here *d*, λ and *B* are tuning parameters, which would determine the performance of the boosting method, and we are going to set d = 1 and use cross-validation to determine the optimal value of λ and B in the following model fitting part.

4.6.2 Model fitting

First of all, we use the **gbm.step()** function in the **R** package **dismo** to determine the number of trees needed for this this problem by a 10-fold cross-validation [18]. This algorithm identifies the optimal number of trees as that at which the holdout devianceis minimized [19]. As a result, the optimal number of trees we get is 10000. After that, we optimize the learning rate by setting the number of trees as 10000 using 5-fold cross-validation to minimize the CV error. The optimal learning rate we get is 0.03.Thus, we decide to set B = 10000 and λ = 0.03 to train our final boosting model by using the training data.

5. Model summary

5.1 Model comparison

5.1.1 Combined ROC curves

Since this is a binary classification problem, to assess the performance of several models in Section 4, we consider to use receiver operating characteristic (ROC) curves and area under curve (AUC) values to measure the performance of our models. The combined ROC curves of our all six models are displayed in Figure 6.

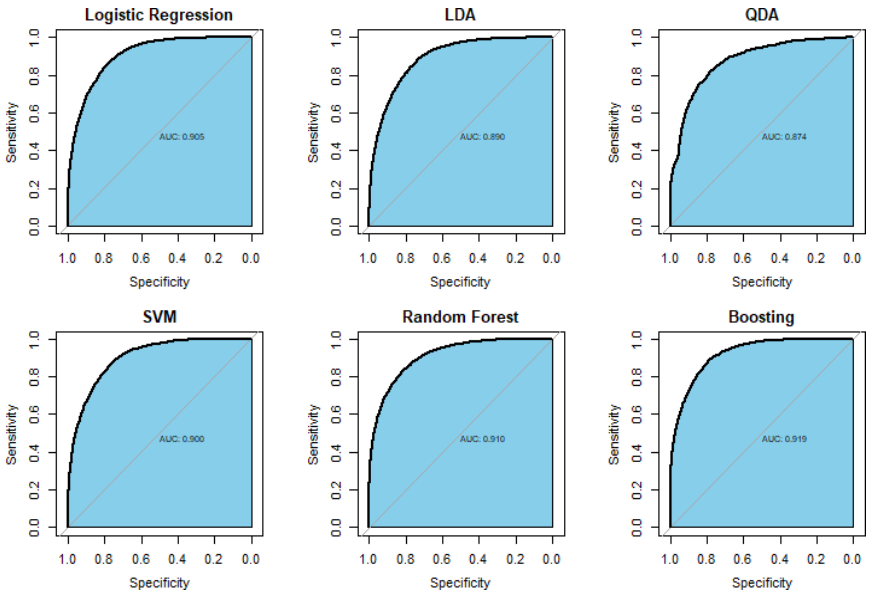


Figure 6. Combined ROC curves

From the ROC curves and the corresponding AUC values of those models, we find that logistic regression, SVM, random forest and boosting have similar classification performance on testing data. Their AUC values are all greater than 0.9, while the boosting model has the highest AUC values of 0.9192. Therefore, we consider choosing our final model among logistic regression, SVM, random forest and boosting.

5.1.2 Prediction performance

Since the response variable **salary** is unbalanced, where 76.1% of individuals have an income level of "≤ 50K" and the other 23.9% of individuals have an income level of " > 50 K", we decide to treat " > 50 K" as the positive level and compute the measures for performance of classification in Table 2, where the definitions of several commonly used criteria, such as accuracy, recall, specificity, precision, and F-measure, can be found in [10] and [20].

Table 2. Comparison of prediction performance

Models	Accuracy	Recall	Specificity	Precision	F-measure
Logistic regression	84.96%	59.71%	92.99%	73.05%	65.71%
LDA	84.03%	57.62%	92.43%	70.79%	63.53%
QDA	76.19%	85.67%	73.18%	50.41%	63.47%
SVM	84.83%	77.82%	86.18%	51.99%	62.34%
Random forest	76.67%	66.27%	93.81%	76.52%	71.03%
Boosting	86.33%	62.31%	88.68%	76.69%	68.76%

From Table 2, we can see that the boosting model has the highest prediction accuracy, while the random forest model has the highest F-measure. However, the prediction accuracy of the random forest model is only 76.67%, which is the second-worst amongall six models, while the boosting model has a second-best F-measure. Therefore, we decide to choose the boosting model as our final model and make further improvements and interpretations based on this model.

5.2 Boosting prediction

When we use this boosting model to make predictions on the test data, the results we get are predicting probabilities. So we need to optimize the cutoff using cross- validation such that the cutoff maximizes the accuracy. By using the cross-validation approach, the optimal cutoff we get is 0.5022067, which is very close to 0.5. So we use this optimal cutoff to make prediction on the testing data and the final confusion matrix we get is shown in Figure 7.

From the confusion matrix, we can see that the predictive accuracy of our final modelis about 86.34%, which is a bit higher than the accuracy by using default cutoff 0.5.

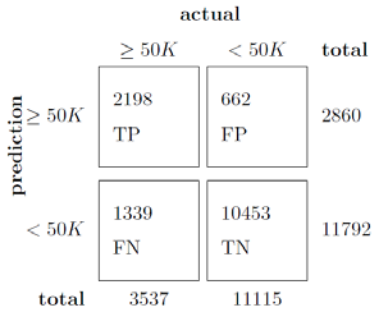


Figure 7. Confusion matrix

5.3 Interpretation

According to the boosting model, we summarize the relative importance of each variable as shown in Figure 8.

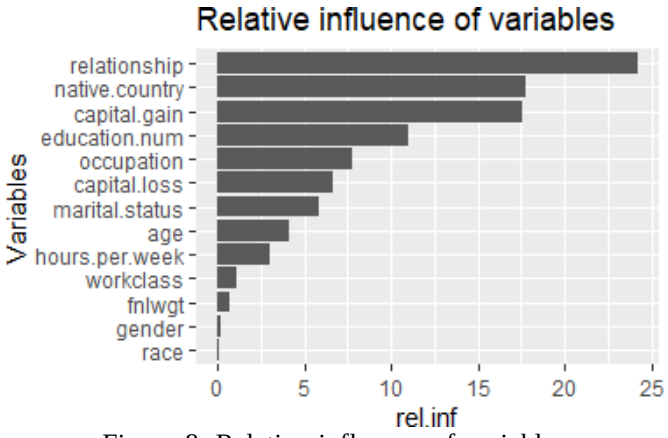


Figure 8. Relative influence of variables

From Figure 8, we can see that **relationship**, **native.country** and **capital.gain** are the three most important variables that determine the level of adult income. It can be interpreted that the income level of an individual is mainly decided by his/her relationship in a family. Besides, where the individual comes from and how much capital he/she has also have a big influence on his/her income level.

Moreover, we also fit a decision tree model to explain the data intuitively. The decision tree model is displayed in Figure 9.

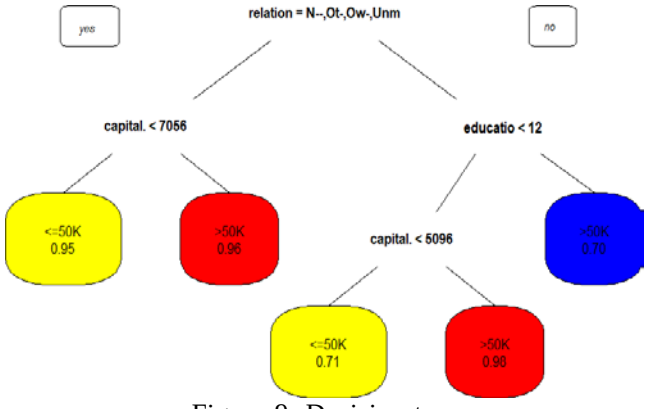


Figure 9. Decision tree

From Figure 9, we can find that for educated (*education.num* ≥ 12) married individuals (*relation* = *Husband, Wife*), their income tends to be at a higher level. But for married individuals who have not received much education, their income level is highly correlated with their **capital gain**. Besides, the income of those un-married individuals (relation = Not-in-family, Other-relative, Own-child, Unmarried) is mainly determined by their **capital gain**.

6. Conclusion

In this paper, we explore the factors that affect the income of adults. We apply logistic regression, LDA, QDA, SVM, random forest and boosting to solve thisproblem. After comparing the performance of these models, we find that boosting provides the highest AUC value, which indicates that it is the best model for fitting the data. Moreover, we find **relationship**, **native.country**, and **capital.gain** have the largest impact on salary levels.

Although we solved the problem successfully, we still have some extensions. For example, most variables in this dataset are categorical and some of them have many levels. We think it would improve the performance of our models by using some suitable methods to reduce the number of levels for some categorical predictors before constructing predicted models. Besides, the performance of LDA and QDA is bad because the assumption of normality is invalid. Some methods can be employed to transform variables into normal distribution, which we do not consider in this project. Moreover, more advanced data science methods like XG-boost and neural network [21-23] or Artificial Intelligence approaches [24] can also be applied to deal with this problem, and they might have a better prediction performance than the current models. In addition, we only consider the relationship between salary and the other covariates in this project. But the relationship between covariates might also be interesting, we can adopt graphical models [25] to look into the dependence between predictors.

7. References

[1] Kohavi R, Becker B. Adult data set. 1996. https://archive.ics.uci.edu/ml/ datasets/adult//

[2] Olafenwa J. Us adult income. 2017. https://www.kaggle.com/johnolafenwa/us-census-data/kernels/

[3] Chakrabarty N, Biswas S. A statistical approach to adult census income level prediction. 2018. arXiv: 1810.10076.

[4] Topiwalla M. Machine learning on UCI adult data set using various classifier algorithms and scaling up the accuracy using extreme gradient boosting. university of sp jain school of global management.

[5] Lazar A. Income prediction via support vector machine. International Conference on Machine Learning and Applications - ICMLA 2004. 16-18 December 2004. Louisville, KY, USA.

[6] Deepajothi S, Selvarajan S. A comparative study of classification techniques on adult data set. International Journal of Engineering Research Technology (IJERT). 2012; 1(8).

[7] Hastie T, Tibshirani R, Friedman J. The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer, New York. 2008.

[8] Hastie T, Tibshirani R, Wainwright M. Statistical Learning with Sparsity: The Lasso and Generalizations. CRC Press, New York. 2015.

[9] Johan AK, Suykens Bart Hamers, Bart De Moor. Coupled transductive ensemble learning of kernel models. 2003. https://www.researchgate.net/publication/228542462_Coupled_transductive_ensemble_learning_of_kernel_models/

[10] Chen L-P, Yi GY, Zhang Q, He W. Multiclass analysis and prediction with network structured covariates. Journal of Statistical Distributions and Applications. 2019; 6:6. DOI: 10.1186/s40488-019-0094-2.

[11] Chen L-P. Model-based clustering and classification for data science: with application in R by Harles Bouveyron, Gilles Celeus, T. Bredan Murphy and Adrian E. Raftery. Biometrical Journal. 2020; 62: 1120-1121. DOI: 10.1002/bimj.201900390.

[12] Chen L-P. Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar: Foundations of machine learning, second edition. Statistical Papers. 2019; 60(5): 1793–1795.

[13] James G, Witten D, Hastie T, Tibshirani R. An Introduction to Statistical Learning. Springer, New York. 2013.

[14] Breiman L. Classification and Regression Trees. Routledge, New York. 1984.

[15] Robert J, Tibshirani BE. An Introduction to the Bootstrap. Chapman Hall/CRC, New York. 1993.

[16] Quinlan JR. Induction of decision trees. Machine Learning. 1986; (1).

[17] Quinlan JR. C4.5: Programs for Machine Learning. Morgan Kaufmann. 1992.

[18] Jane Elith, John Leathwick. Boosted regression trees for ecological modeling. R Documentation. Available online: https://cran. r-project. org/web/packages/dismo/vi- gnettes/brt. pdf (accessed on 12 June 2011), 2017.

[19] Robert J Hijmans, Steven Phillips, John Leathwick, Jane Elith, Maintainer Robert J Hijmans. Package 'dismo'. Circles. 2017;9(1):1–68,.

[20] He W, Yi GY, Chen L-P. Support vector machine with component graphical structure incorporated. Proceedings, Machine Learning and Data Mining in Pattern Recognition, 15th International Conference on Machine Learning and Data Mining, MLDM 2019, vol.II. 2019: 557-570.
[21] Chen L-P. Using machine learning algorithms on prediction of stock price. Journal of Modeling and Optimization. 2020; 12: 84-99. DOI: 10.32732/jmo.2020.12.2.84
[22] Chen L-P, Zhang Q, Yi GY, He W. Model-based forecasting for Canadian COVID-19 data. PLOS ONE, 2021; 16(1): e0244536. DOI: 10.1371/journal.pone.0244536.
[23] Chen L-P. Practical Statistics for Data Scientists: 50+ Essential Concepts Using R and Python by Peter Bruce, Andrew Bruce, and Peter Gedeck. Technometrics. 2021; 62, 272-273. DOI: 10.1080/00401706.2021.1904738.
[24] Chen L-P. Artificial Intelligence for Drug Development, Precision Medicine, and Healthcare by Mark Chang. Biometrics. 2020; 76: 1392-1394. DOI: 10.1111/biom.13390
[25] Chen L-P. Multi classification to gene expression data with some complex features. Biostatistics and Biometrics Open Access Journal. 2018; 9(1): 555751. DOI: 10.19080/BBOAJ.2018.09.555751.