Salary Prediction of a Person based on Adult Dataset

Venu Babu Kolli[1], Rashmi Mallappa[2], Pratik Magar[3]

[1]Department of Applied Data Science, School of Informatics and Computing, IUPUI
[2]Department of Computer and Information Sciences, School of Science, IUPUI
[3]Department of Mechanical Engineering, School of Engineering and Technology, IUPUI

ABSTRACT

This paper is focused on building a predictive model on adult dataset by implementing various machine learning algorithms which can accurately identify individuals whose salary exceeds a specified value. Applying the concepts of machine learning algorithms such as Logistic Regression, Support Vector Machines, K-Nearest Neighbors and Neural Networks on Adult dataset, the hidden patterns can be derived by building a predictive model. Upon building the models, the performance of the model is evaluated, and the efficiency of the algorithms applied on this dataset are compared.

Keywords – Intelligent systems, Machine learning, Neural networks, Logistic Regression, SVM, K-Nearest Neighbors, predictive model, Adult data.

INTRODUCTION

Many data mining tasks require classification of data into classes. For example, loan applications can be classified into either 'approve' or 'disapprove' classes. A classifier maps data items into one of the several classes. This project will investigate the data mining of demographic data in order to create one or more classification models which are capable of accurately identifying individuals whose salary exceeds a specified value. The data used in this project were sourced from the University of California Irvine data repository and are referred to as the adult dataset and contain information on individuals such as age, level of education and current employment type of predicting whether the income is ≥ 50k/year from a person's attributes, by using important features.

Problem Statement

The aim of this project is to implement various machine learning algorithms to arrive at a model that works best for the predictive task by carrying out performance analysis to predict whether an individual's annual income exceeds 50k/year or not. The goal is to apply the classification data mining tool to the US census data to profile the variables that affect US household incomes. 50K was used as a base line for middle class income and investigated how the different attributes of a household contributes to the probability of that person being classified as earning equal to or more than middle class income or not.

DATASET

Data Set Description

Data set consist of 32,561 rows of data and 15 related attributes. There are three different type of data in the dataset, 4 continuous attributes, one binomial and 10 discrete attributes. The binomial label indicating a salary of less or greater than fifty thousand US dollars, which for brevity, will be referred to as <50K or >50K in this project. The work class attribute describes the type of employer such as self-employed or federal, occupation describes the employment type such as sales, tech-support. The education attribute describes the highest level of education attained by individual such as high school graduate or

doctorate. The relationship attribute has categories such as husband, wife or not-in-family and the marital status attribute has categories such as married, divorced or separated. The final nominal attributes are native country, gender and race. The continuous attributes are age, hours worked per week, education number, capital gain and loss.

Data Collection

The Adult dataset is taken from University of California, Irvine (UCI) Machine Learning Repository. The following is a link to the dataset: https://archive.ics.uci.edu/ml/datasets/Adult

METHODOLOGY

The project is broken down into five different phases. In first phase, the dataset was collected, and pre-processing was done on the dataset. In the second phase, feature selection method is applied to identify best set of features for prediction. Then splitting of data into test and train datasets. In third phase, the predictive model was built based upon algorithms like Neural Networks, Logistic Regression, SVM classification, K-Nearest Neighbors and the salary of an adult was predicted. In the last phase, performance of each algorithm was analyzed based on predicted values and actual values, using confusion matrix. Python programming was used as a platform to perform all of the project tasks.

DATA PREPROCESSING

The acquired dataset from the UCI was raw data which had null and redundant data values as well. In order to ensure better results in the prediction process this raw data was converted into structured data. All the null values and duplicate data in the dataset were removed thus improving the quality of the data.

Feature Selection

Feature selection is a process of finding right set of attributes for predicting dependent variable. The variables which are irrelevant and redundant, which have almost no adverse effect on the dependent variable are removed in the feature selection process. The feature selection process improves efficiency of the predictive model by minimizing dimensionality and hence, the time taken for the model to be built can be reduced. [11]
In Recursive feature elimination, each feature is ranked by the model's coef_ or feature_importances_attributes, and in each iteration, the feature with lowest importance score is eliminated until no input is left. [11]

Splitting of Data

The dataset was divided into training data and testing data. The training data is the actual data that is used to train the predictive model and the model learns from this data. Testing data is used to test the trained model and evaluate the predictive model. The data was divided in the ratio 75:25 in order to train the model accurately and to assess the performance of the predictive model accurately.

LOGISTIC REGRESSION CLASSIFIER

Logistic Regression is a predictive analysis method which is used to describe data and explain the relationship between one dependent binary variable and one or more independent variables. This type of analysis is appropriate to conduct only when the dependent variable is binary (dichotomous). [5]

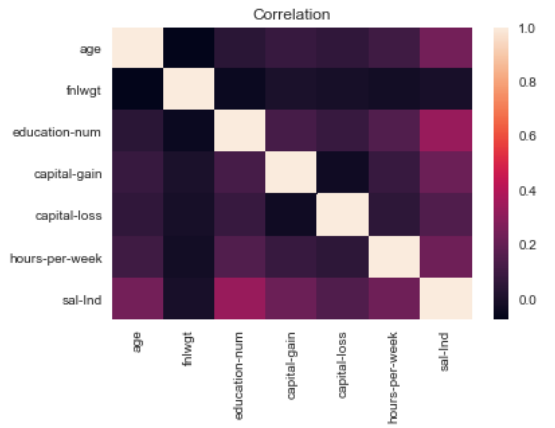
Some of the assumptions made in logistic regression analysis are as follows -

1. The dependent variable should be dichotomous in nature (present or absent)

2. There should be no outliners in the data
3. There should be no high correlations among the predictors [10]

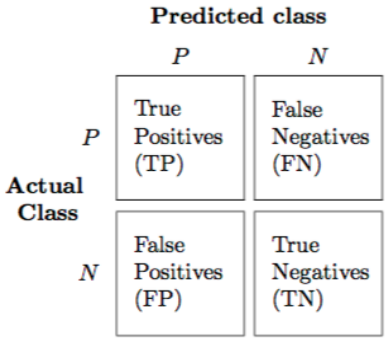
In the dataset, the dependent variable is dichotomous, i.e. either ≤ 50k or > 50k. During the data preprocessing, the data entries with null value variables and outliners was eliminated from the dataset.

The correlations matrix was found to check for high correlation among the predictor variables. This matrix is showing in the figure. It can be seen that the correlation among the predictor variables is very low. Hence this data now can be used for training the logistic regression model.



(Fig. 1 – Correlation Matrix)

For training the logistic regression model, 75% of the total dataset was used. The remaining 25% of the dataset was used to check if the model can predict the data accurately. A good way to know the accuracy of the model is to generate a confusion matrix (error matrix). In a confusion matrix, the number of correct and incorrect predictions are summarized with count values which are broken down by each class. It is a very important tool to get an insight of the errors being made by the classification model and also to know the types of errors being made. [3] A typical confusion matrix is shown in following figure.



(Fig. 2 – Confusion Matrix Format [3])

The prediction confusion matrix obtained from logistic regression model is shown in the following figure.

```
[[5231  397]
 [1007  906]]
```

(Fig. 3 – Confusion Matrix for Logistic Regression Classifier)

This states that out of the total predicted data, 5231 were actually positive and were predicted positive as well. 906 were actually negative and were predicted negative as well. From the remaining data, 1007 were wrongly predicted as positive when they were actually negative and 397 were wrongly predicted as negative when they were actually positive.

The accuracy of the logistic regression model was found out to by using the formula,

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Where,

TP = True Positive Values

TN = True Negative Values

FP = False Positive Values

FN = False Negative Values

The accuracy of Logistic Regression Model was found to be 81%.

Recall, Precision and F-Score

As accuracy assumes equal cost for both kinds of errors, it cannot be used as an only measure to conclude the effectiveness of the classification model. For this reason, Recall, Precision and F-Score are used to further validate the model. [8]

Recall is the ratio of total number of correctly classified positive examples to the total number of positive examples. The correct classification of a class can be indicated by a high recall value. Recall is given by the relation, [8]

$$Recall = \frac{TP}{TP + FN}$$

Precision can be defined as a ratio of total number of correctly classified positive examples to the total number of predicted positive examples. High precision indicates that an example labelled as positive is actually positive. Precision can be calculated as, [8]

$$Precision = \frac{TP}{TP + FP}$$

To represent both the recall and precision, they can be expressed together using the F-Score. F-Score is a harmonic mean of the recall and precision. This is calculated as, [8]

$$F\ Score = \frac{2 * Recall * Precision}{Recall + Precision}$$

The Precision, Recall, and F-Score obtained are shown in the following figure.

	precision	recall	f1-score	support
0	0.84	0.93	0.88	5628
1	0.70	0.47	0.56	1913
avg / total	0.80	0.81	0.80	7541

(Fig. 4 – Precision, Recall and F-Score for Logistic Regression Classifier)

Receiver Operating Characteristics

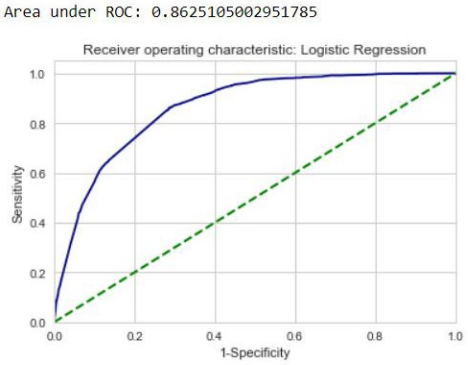
To calculate the efficiency of the binary classifier such as logistic regression, a Receiver Operating Characteristic (ROC) curve is plotted. The ROC curve is found out by plotting Sensitivity (True Positive Rate) on y-axis against the Specificity (True Negative Rate) on x-axis.

Here, Sensitivity measures the proportion of positives that are correctly identified as such and Specificity measures the proportion of negatives that are correctly identified as such. [1]

$$Sensitivity = \left(\frac{TP}{TP + FN}\right) * 100$$

$$Specificity = \left(\frac{TN}{TN + FP}\right) * 100$$

The Receiver Operating Characteristics of Logistic Regression Analysis was found out and is shown in the following figure.



(Fig. 5 – ROC Curve)

SUPPORT VECTOR MACHINES CLASSIFIER

A support vector machine (SVM) is a discriminative classifier which was defined by a separating hyperplane in an N-dimensional space. For supervised learning, if the algorithm is given a labelled data, it can output an optimal hyperplane. This hyperplane is considered as a decision boundary which can help classify the data points. Data points that fall on either side of the hyperplane can be attributed to different classes. The dimension of the hyperplane is dependent on the number of features. [7]

Support vectors are data points that are closer to the hyperplane and influence the position and orientation of the hyperplane. It is possible to maximize the margin of classifier using these support vectors. [7]

Similar to Logistic Regression, the output of Support Vector Machines analysis was a confusion matrix as follows –

```
[[5072  556]
 [ 865 1048]]
```

(Fig. 6 – Confusion Matrix for SVM Classifier)

This states that out of the total predicted data, 5072 were actually positive and were predicted positive as well. 1048 were actually negative and were predicted negative as well. From the remaining data, 865 were wrongly predicted as positive when they were actually negative and 556 were wrongly predicted as negative when they were actually positive.

The accuracy of the support vector machine analysis was found out to be 81%

The Precision, Recall and F-Score were found out as follows –

	precision	recall	f1-score	support
0	0.85	0.90	0.88	5628
1	0.65	0.55	0.60	1913
avg / total	0.80	0.81	0.81	7541

(Fig. 7 – Precision, Recall and F-Score for SVM Classifier)

K-NEAREST NEIGHBORS CLASSIFIER

A K-Nearest Neighbor (KNN) algorithm is used for both classification and regression prediction problems. But it is more widely used in classification problems in the industry. KNN is a non-parametric technique which does not make any assumptions on the underlying data distribution. Hence, the model structure is determined from the data. [6]

In this algorithm, an object is classified by a majority vote of its neighbors, with the object being assigned to the class most common among its K-nearest neighbors. The value of K is used for minimizing the training and validation error rate. An optimal value of K can help in achieving accurate results. For this analysis, a K value of 3 is selected. [6]

The algorithm was implemented using Python and the output of KNN was found out to be a confusion matrix as follows –

```
[[4336 1292]
 [ 708 1205]]
```

(Fig. 8 – Confusion Matrix for KNN Classifier)

This states that out of the total predicted data, 4336 were actually positive and were predicted positive as well. 1205 were actually negative and were predicted negative as well. From the remaining data, 703 were wrongly predicted as positive when they were actually negative and 1292 were wrongly predicted as negative when they were actually positive.

The accuracy of K-Nearest Neighbors was found out to be 0.73. This was the lowest accuracy among the three machine learning algorithms used.

The Precision, Recall and F-Score were found out as follows –

	precision	recall	f1-score	support
0	0.86	0.77	0.81	5628
1	0.48	0.63	0.55	1913
avg / total	0.76	0.73	0.75	7541

(Fig. 9 – Precision, Recall and F-Score for KNN Classifier)

NEURAL NETWORKS

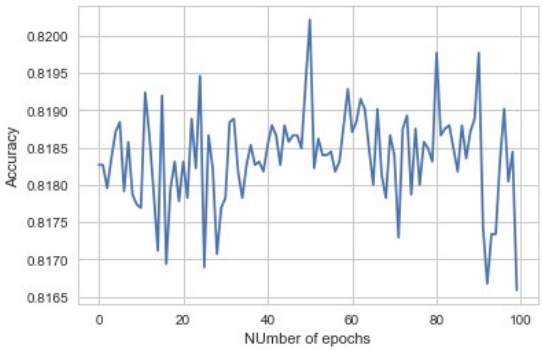
The concept of Neural Networks is closely related to the structure of a human brain. A human brain contains a dense interconnected network of neurons. In similar manner, neural networks are built out of a densely interconnected set of units. Each unit takes a number of inputs and produces a single output. Neural network learning methods are robust to noise in the training data and the final output does not get affected by the errors in training data. Another advantage of using neural networks is that they are able to bear longer training times which depend on factors such as weights in the network, number of training examples provided, and the setting used for various learning algorithm parameters. [4]

Neural Networks can be of two types –

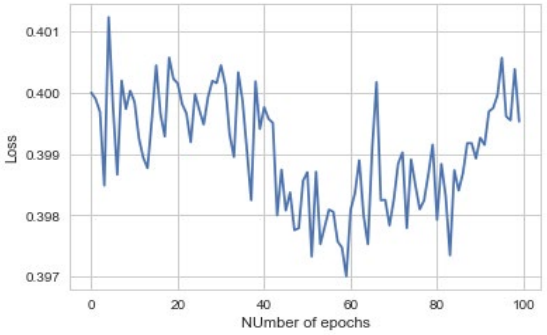
1) Single Layer Neural Network

2) Multi-Layer Neural Network

This algorithm was implemented using one input layer, two hidden layers and output layer. Sigmoid activation function is used in the final output layer as the classification variable is dichotomous. This activation function will provide the probability of being the class either 0 or 1. [9]



(Fig. 10 – No. of Epochs vs. Accuracy)



(Fig. 11 – No. of Epochs vs. Cost Function)

The above model is trained over 100 epochs with batch size of 10 chosen randomly from the training data. From the Fig. 10 as the number of epochs increases the accuracy increases and at 50 epochs the maximum accuracy is achieved. From the Fig. 11 as the number of epochs increases the cost function (loss) decreases till the cut-off point is reached. The cut-off point is where training beyond this will result in overfitting of model and again loss will be increased on the test data.

The performance of model on test data the neural networks based on the below confusion matrix is as follows

```
[[3751 1877]
 [ 240 1673]]
```

(Fig. 12 – Confusion Matrix for Neural Networks)

This states that out of the total predicted data, 3751 were actually positive and were predicted positive as well. 1673 were actually negative and were predicted negative as well. From the remaining data, 1877 were wrongly predicted as positive when they were actually negative and 240 were wrongly predicted as negative when they were actually positive.

The accuracy of Neural Networks was found out to be 0.825. This was the highest accuracy compared to the other three machine learning algorithms used.

RESULTS

Algorithm	Accuracy
Logistic Regression	81
Support vector classifier	81
K-Nearest Neighbors Classifier	73
Neural Networks	82.5

(Table 1 – Obtained Accuracy of all Algorithms)

From the results, it is seen that K-Nearest Neighbors Classifier provides the least accuracy (73%) and the Neural Networks provides the maximum accuracy (82.5%).

CONCLUSION

From the results, it can be concluded that for the dataset selected, Neural Networks give the best result. Hence, it can be said that for the selected dataset, Neural Networks is the best algorithm for salary prediction based on demographics of a person.

REFERENCES

1. Andrew P. Bradley, “The use of area under the ROC curve in the evaluation of machine learning algorithms”, Pattern Recognition, Volume 30, Issue 7, July 1997

2. Nigel Williams, Sebastian Zander, Grenville J. Armitage, “A preliminary performance comparison of five machine learning algorithms for practical IP traffic flow classification”, Center for Advanced Internet Architectures, Swinburne University of Technology, October 2006.

3. https://en.wikipedia.org/wiki/Confusion_matrix [Online]

4. https://scikit-learn.org/stable/modules/neural_networks_supervised.html [Online]

5. https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html [Online]

6. https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html [Online]

7. https://scikit-learn.org/stable/modules/svm.html [Online]

8. https://www.geeksforgeeks.org/confusion-matrix-machine-learning/ [Online]

9. https://en.wikipedia.org/wiki/Sigmoid_function [Online]

10. https://www.statisticssolutions.com/logistic-regression-assumptions/ [Online]

11. http://www.scikit-yb.org/en/latest/api/features/rfecv.html [Online]