

Income Prediction

Name:	Udaybhan Rathore
Registration No./Roll No.:	19328
Institute/University Name:	IISER Bhopal
Program/Stream:	DSE
Problem Release date:	February 02, 2022
Date of Submission:	24/04/2022

1 Introduction

In recent times, it has become clear that the applications of machine learning go beyond science and technology. As governments grapple with the issues of wealth inequality and poverty, machine learning can be used as a tool in seeking solutions to these problems. The 1994 Census Bureau Database provides an extensive look into various descriptors of the population like Age, workclass, fnlwgt, eductoin, eductoin num, marital status, occupation, relationship, race, sex, capital.gain, capital.loss, hours.per.week, and native.country, with Boolean logic for each sample describing whether or not each person makes over or under 50,000 dollars per year. So it is a binary classification problem. we have used sklearn libraries supervised model including KNN, Random forest, logistic regression, Decision tree[1], SVM[2], and compare them on the basis of their F1 measure.

1.1 Dataset

The dataset has 43597 rows and 14 attributes (6 numeric and 8 categorical) which show various anonymised information about a person and one target column which shows whether or not that person's income exceeds 50,000 dollars. Figure 1 show some of the attributes of dataset varying with target variable. The dataset have null values in workclass, occupation, native-country. Income dataset. The dataset is unbalanced, as the dependent feature 'income' contains 76.07 percent values have income less than 50k and 23.93 percent values have income more than 50k.

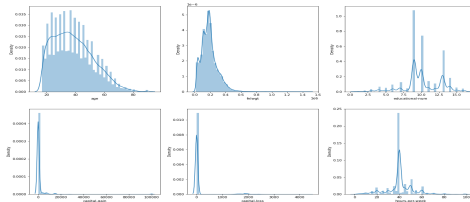


Figure 1: distribution of numerical values

1.2 Data Preprocessing

There are three main data pre-processing steps that we have noted needs to be carried out before we fit our train set to the machine learning models: **imputing missing values**, **categorical attributes**

encoding and feature scaling. We have created a pipeline for the data pre-processing steps. Since our missing values are all in the categorical column, first we create a pipeline to impute the missing values and then one hot encode the columns. For the imputer strategy, since categorical values don't have a mean or median, we are left with the mode (most frequent). A column transformer is then constructed which transforms the categorical column using the earlier constructed pipeline and scales the numerical columns. We have also defined three function confusion-Matrix for evaluation, Grid-Search for splitting the data and for hyper tuning, for prediction. The train set is passed through this pipeline and we are ready to train our model.

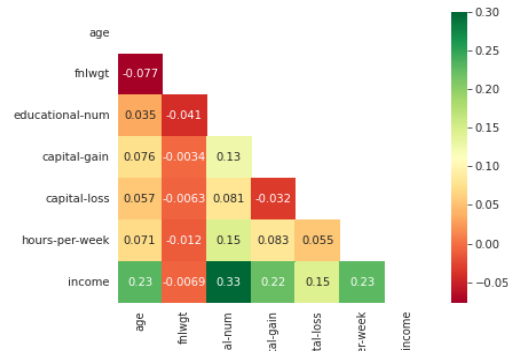


Figure 2: Heat map

2 Methods

The machine learning task is a binary classification type task and so models suitable for classification have been chosen. The models that have been used are Logistic Regression, Decision Tree Classifier, Random Forest, KNN Classifier, Bernoulli Naive Bayes Classifier and Support Vector Machines. We have experimented this model to try to increase the f1 score. The experiments with the machine learning models were carried out in the following order:

- The models were fitted on the train set to see how they performed.
- We have created the pipeline to see best parameters
- The model was fine-tuned again using the new pipeline and new parameters were chosen based on the best parameters gotten earlier.
- The evaluation metrics used are the accuracy F1 score and confusion matrix

github link

3 Evaluation Criteria

Model accuracy alone may not be enough to evaluate a model. Hence the following model performance measures have been used to evaluate the models, based on the confusion matrix built for the predictions on the training and test datasets. Since dataset is not balanced therefore accuracy is not only the correct way to evaluate the model. So we have followed the f1 score, precision, and recall to evaluate our models. True positive tells that model have correctly classified the test dataset that income less than or equal to 50k and true negative tell the model have correctly classified that is income is greater than 50k. ConfusionMatrix for the same is given in 3

Table 2: Model evaluation without Hyper tuning

Model Name	F-measure
KNN	83.97
RF	93.28
SVM	83.47
DT	92.01
BNBC	79.55
Logistic Regression	82.55

Table 1: Model evaluation without Hyper tuning

Classification	Precision	Recall	F-measure
KNN	0.83	0.98	0.90
RF	0.86	0.98	0.91
SVM	0.73	0.57	0.64
DT	0.85	0.87	0.85

4 Analysis of Results

Results of descriptive analysis are as follow.

- the Dataset shows that most of the people work around 40 hours per weeks.
- self-employment peoples have higher probability of getting salary $\geq 50k$
- the Dataset shows that most of the people work around 40 hours per weeks.
- We can infer that doctorate and prof-school educated people have more probability of getting salary of greater than 50k
- The correlation between income and working hours per week of the person positive correlation i.e as working hours per week increases the probability of having salary greater then 50k increases
- In this dataset, the most number of people are young, white, male, high school graduates with 9 to 10 years of education and work 40 hours per week.
- From the correlation 2, we can see that the dependent feature 'income' is highly correlated with age, numbers of years of education, capital gain and number of hours per week.

Result of different machine learning models: Results of different models and there hyper tuning are show in1 and 2

- The best parameter of different models after hyper tuning are a follow for Random forest .The best parameter of random forest model after hyper tuning is when Best score: 'max_depth': 20, 'max_features': 20, 'n_estimators': 500

Table 3: Confusion Matrices of LR and MNB Respectively Using All Terms

Actual \ Predicted Class	0	1
0	TP	FN
1	FP	TN

- The best parameter of KNN model after hyper tuning is when Best parameters: 'knn_neighbors': 50, 'knn_p':1'knn_weights': 'uniform'
- The best parameter of SVM after hyper tuning is when 'gamma': = 'scale', Best parameters: 'gamma': 'scale', 'kernel': 'linear'

You may create different tables e.g., Table 1, Table2 2, Table 2 etc. to show the experimental results. Moreover, you should explain (in simple sentences) your observations from these tables. Therefore analyze the results based on these observations as discussed in the class.

5 Discussions and Conclusion

We have performed, various experiments were conducted using 3 machine learning algorithms on the given Income dataset to predict income classes. The experiments showed that Random Forest performed best with a test accuracy of 93 percent and F1 score of 93.28. This study shows that machine learning can be used to identify factors that cause disparity in incomes and with this knowledge, the correct steps can be taken to bridge the income divide. In terms of future work, more features and attributes can be added to those already included in the dataset and we can see the effects of these features on the accuracy of model prediction. Also other advanced machine learning techniques can be used and their accuracy compared to the models that have been used in this paper.

6 Contribution

Pushpendra Dhakar is a very help and team working man. We together have successfully completed this project. He has done univariate analysis, help in creating a function for hyper tuning, and experiment the different parameter tuning to get the best parameter.

References

- [1] Sisay Menji Beken. Using decision tree classifier to predict income levels. 2017.
- [2] Alina Lazar. Income prediction via support vector machine. In *ICMLA*, pages 143–149. Citeseer, 2004.