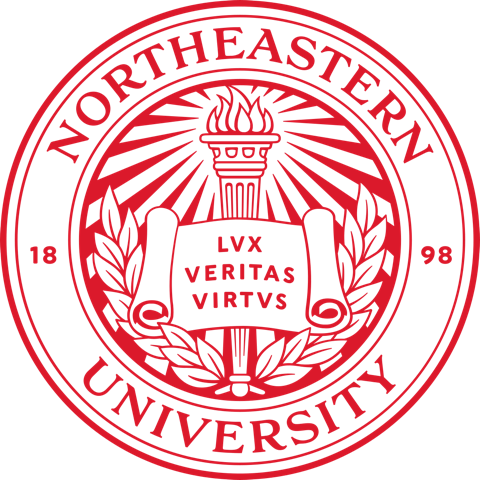
**Module 1 Project**

**Understanding Income Inequality**

Sunil Raj Thota



College of Professional Studies, Northeastern University ALY 6020 – Predictive Analytics

Prof. Justin Grosz

November 06, 2021

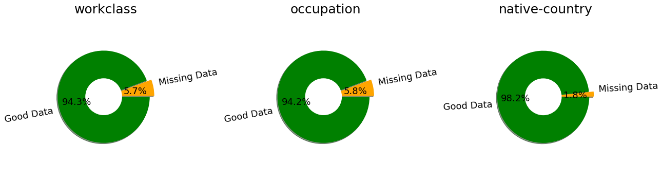
**Introduction:**

Inequalities in wealth and earnings are a big issue, particularly in the United States. One reasonable motivation to lessen the world's rising degree of economic disparity is the possibility of reducing destitution. The notion of universal moral equality promotes long-term growth and improves a country's economic security. Officials in several nations have been working hard to resolve this concern and find the best answer possible. Based on a set of variables, classification has been done to forecast whether a person's annual income in the United States fits into the income categories of greater than 50K Dollars or less than 50K Dollars.

**Part 1:**

**Data Quality:**

To extract certain information from the dataset, the '?' sign is substituted by "NaN" during data filtering. Since the symbol has been changed with NaN, which stands for a null value, these can now be easily identified and computed to verify the sum of the missing values in the collection. There are 48842 records and 15 columns in the census dataset. During my investigation of the dataset, I discovered that the workclass column had 2799 null entries, accounting for 5.73 percent of the total rows in that column. In addition, there are 2809 missing entries in the occupation column, accounting for 5.75 percent, and 857 missing values in the native-country column, accounting for 1.75 percent. The describe function is used to check the dataset's statistics, such as mean, median, mode, and quartile values.

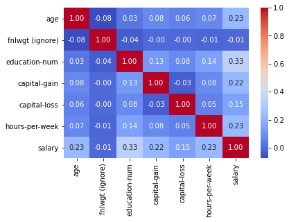


**Data Cleansing and Preprocessing:**

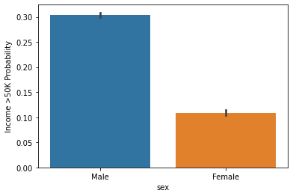
The missing data is now dealt with following specified rules. We'll utilize the "Replace by Frequency or Mode" method because we only have categorical missing data. Without any numerical precedence, convert categorical variables to continuous variables (Label Encoding with Binarization). We have 9 categorical variables. To transform category variables into label encoded binary values, we used the pd.get dummies method. Positive verification is indicated by 1; negative verification is shown by 0. Dummy variables, also known as indicator variables, take discrete values such as 1 or 0 to indicate the existence or absence of a specific category. To avoid the Dummy Variable Trap, remove one column. We removed the "Never-worked" field and used it as the default value. The multi-collinearity was reduced in this case. All additional columns were encoded in the same way.

**Exploratory Data Analysis:**

The goal of this report is to use the US Census dataset to test among the most common supervised machine learning models, the k-NN/ k-Nearest Neighbor approach. The dataset's property Salary is a parameter that will be forecasted using a k-NN algorithm. It is a classification task, in which we must determine to see how well we can distinguish between low- and high-income citizens based on some inputs to our classifier.



The process of rescaling one or more attributes to have a mean value of 0 and a standard deviation of 1 is known as data standardization. Because there are more observations with Income = 50K than with Income >50K, the observation is strongly skewed. An education-num with more than 14 has a greater probability of getting > 50 k income. Also, 40 to 60 hours per week are earning more probable to get paid above 50 K salaries. 25 to 45 is the age where people are earning more.



Males are 30 % probable in getting a higher paycheck. Relationships having as husband or Wife are earning more. By subtracting the mean and scaling to unit variance, Sklearn’s s standard scalar preprocessor is used to Standardize features. By computing the necessary statistics on the samples in the training set, each feature is individually centered and scaled. The mean and standard deviation are then saved and utilized to transform later data.

**Part 2:**

**Data Modelling:**

Because of its simplicity and accuracy, K-NN, or K-Nearest Neighbors, is one of the most well-known classification algorithms in use today. KNN has been utilized in pattern recognition and statistical estimation. It classifies to assign the most common level among the K training samples that are closest to the query point's value. When the machine discovers new data, it looks for value within or near the boundary. If the value of k is low, the noise has a significant impact on the output. The value of K has a significant impact on KNN's performance. The K-NN method calculates commonalities among each occurrence, in this example each census data and its metrics, to produce forecasts.

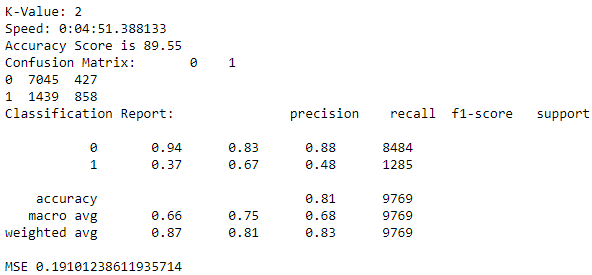
The method compares the new data point to other points in the training data or the data on which the model was created by computing the distance between them. Different distances can be estimated, but the most common is the Euclidean distance, which is useful for data with lesser dimensions. We were given census data on the characteristics of US residents (occupation, education, gender, race). The dataset includes information on people's employment status (self-employed, state government, etc. ), an education level (Bachelors, Masters, High school graduate, etc. ), marriage status (married, widowed, etc. ), occupation, relationship status, gender, race, and income level. To classify low-income residents from high-income citizens, I will model the data using income as the goal variable. This model will also assist us in determining the impact of other factors on income. The Nearest Neighbors Model is created using the cleaned dataset using the k - Nearest Neighbors approach, with an 80:20 split between the training and test sets. The KNN model was run and validated with k values ranging from [2, 5, 10]. When I compared all of the k-values to the appropriate accuracy score, I discovered that the accuracy score with the k-value of 2 is the highest. This model has a high accuracy rate of 89.5, which is an excellent score. Accuracy score has been in the range of 85 percent to 89 percent with values ranging from [2, 5, 10]. As a result, in terms of k values, these accuracy values do not deviate significantly from their range. This demonstrated that this model may be employed in the actual world as a balanced model.

**Findings and Recommendations:**

I have observed that the most important feature is age, followed by capital, education, and hours per week. I will determine 2 as our final k-value. Because, it provides the best accuracy in low time from the evaluations and calculations made using the accuracy score, confusion matrix, MSE, and Speed. For our dataset, we'll need to pick a k-value that strikes a balance between the evaluation procedure's computational cost (not too many model evaluations) and an impartial estimate of model performance. To test the result, we might utilize cross-validation, which is a more stable method. For cv=5, here is the accuracy output



We rotate our validation set and use the remainder of the data to train, for example, by splitting the data into K folds, training on (K–1) folds, and testing on 1 fold as the validation set, a technique known as K-fold cross-validation.

****

For the k-value of 2, the time took to run the model is 4.51 mins, and obtained an accuracy score of 89.55%. From the confusion matrix, there are 7045 TP, 427 FN, 1439 FP, and 858 TN. The MSE value is 0.191.

The ultimate accuracy is calculated by averaging all of the errors. As a result, I would consider this model to be well-balanced. For the k-value of 5, the time took to run the model is 8.49 mins, and obtained an accuracy score of 87.58%. From the confusion matrix, there are 6782 TP, 690 FN, 1015 FP, and 1282 TN. The MSE value is 0.174.

For the k-value of 10, the time took to run the model is 5.48 mins, and obtained an accuracy score of 85.68%. From the confusion matrix, there are 6967 TP, 505 FN, 1153 FP, and 1144 TN. The MSE value is 0.169.

The variability of model prediction for a specific data point or value, which tells us about the dispersion of our data, is known as a variance. A high variance model pays close attention to training data and does not generalize to data it hasn't seen before.

**Conclusion:**

This demonstrates that the k-Nearest Neighbor Model, also known as the Lazy Learner Model, is among the easiest supervised machine learning techniques because it uses all the data to train but the model's output is excellent enough to be trusted. For a better mapping of the attributes to our needed target variable, several attributes of the dataset must be transformed or represented in numerical values ( Salary). These values are set using dictionary mapping. This demonstrates that the k-Nearest Neighbor Model is one of the simplest supervised machine learning methods, as it uses all of the data for training and has a good enough performance to be trusted. For the accurate identification of target names(Salary), each data point of each numerical attribute, also known as feature names, is considered by computing the distance using the Euclidean Distance technique. As a result, I would consider this model to be well-balanced.

**References:**

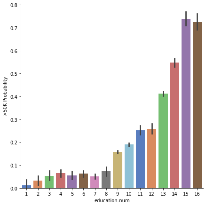
Chauhan, N. (2019). Classifying heart disease using k-nearest neighbors. *KD Nuggets*.

Retrieved from https://www.kdnuggets.com/2019/07/classifying-heart-diseaseusing- k-

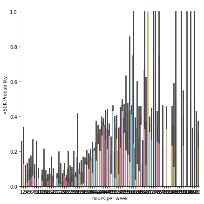
nearest-neighbors.html

**Appendix:**

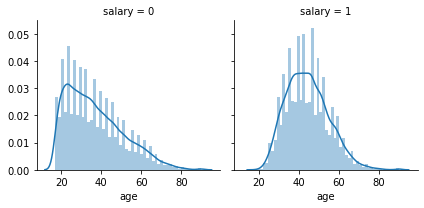
**Figure 1: Education Num vs Salary**

****

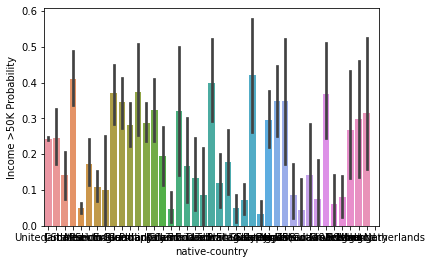
**Figure 2: Hours Per Week vs Salary**

****

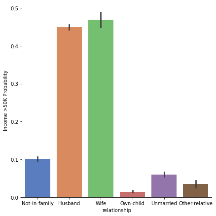
**Figure 3: Age vs Salary**

****

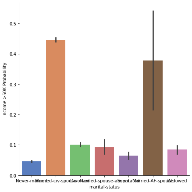
**Figure 4: Native Country vs Salary**

****

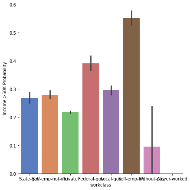
**Figure 5: Relationship vs Salary**

****

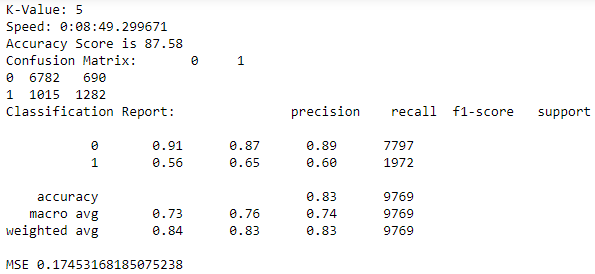
**Figure 6: Marital Status vs Salary**

****

**Figure 7: Workclass vs Salary**



**Figure 8: k-Value = 5**

****

**Figure 9: k-Value = 10**

