*CAP 4630 Optional Project*

*Prediction modeling to determine whether a person makes over $50K a year using machine learning methods from a census dataset*

Patrick Lynch

Dept. of Computer Science

College of Engineering and Computer Science

University of Central Florida, Orlando, Florida, 32816-2450, United States

[lynch\_patrick@knights.ucf.edu](mailto:lynch_patrick@knights.ucf.edu)

ID #4796470

Ricky Egawa

Dept. of Electrical and Computer Engineering

College of Engineering and Computer Science

University of Central Florida, Orlando, Florida, 32816-2450, United States

[rickyegawa@knights.ucf.edu](mailto:rickyegawa@knights.ucf.edu)

ID #4281212

Abstract — Money plays a great role in our collective society as it provides the opportunity for individuals to achieve a better overall quality of life. A stable income allows an individual to have greater opportunities in life such as purchasing a home, opening a business, investing in higher education, and so much more. With the amount of money, a person makes being very valuable information to many corporations, big businesses, and curious individuals to further their respective knowledge and awareness for what’s going on in the world, our goal for the CAP 4630 Optional Project is to create a predictive model to estimate whether an individual has an income exceeding $50,000 a year based on other information about the individual provided by the Census Income dataset. We also propose different ensemble learning strategies that came out to be beneficial while making predictions for our estimation models. The prediction estimation models are made using various machine learning models, methods, and algorithms. The numerous machine learning models, such as Logistic Regression, Random Forest Classifier, Decision Tree Classifier, and Support Vector Machine (SVM) are used in predicting and analyzing the dataset. While numerous other techniques and functions such as StandardScaler, Synthetic Minority Oversampling Technique (SMOTE), and GridSearchCV were all used to help predicting and analyzing the dataset.

Keywords — Individual; Income; $50,000; Prediction; Model; Census Data; Logistic Regression; Random Forest Classifier; Decision Tree Classifier; Support Vector Machine (SVM); Techniques; StandardScaler; Synthetic Minority Oversampling Technique (SMOTE); GridSearchCV

# Introduction

Census is a survey conducted on the full set of observation objects belonging to a given population [7]. In context, a census is a complete collection of a population at a time with respect to defined characteristics, which in this case are individuals.

$50,000 a year of income is an earned value of monetary compensation given by the employer to be received by the employee on a regular basis. With the average annual wage in 2019 in the United States being $51,916.27 [13], the predictive income value for individuals of $50,000 for our project is in the range of plus or minus $2,000.

Census and Income are directly relational to one another. The census data collects high-level statistical data about an individual’s race, families and living arrangements, as well as a person’s health, marital status, education, employment, and housing, etc. As all these collected census data features are being analyzed in real life, a person’s income can also be easily inferred by these features as an individual can easily be influenced and ultimately be affect by these census measures. Overall, an individual’s economic measure and condition, determining whether a person’s income is either low or high according to the data provided compared to the national average can be predicted by the provided features.

Predicting a person’s income using the census data can be beneficial to many individuals. Especially in a scenario where those individuals who may have been determined to have a low income can qualify to participate in government funded programs such as Medicaid, and Food Stamps which can help those individuals in need of financial assistance who are not fortunate enough to afford those services.

The census data contains complete information about the individual’s set of belonging objects in society, which in term will help train the machine and predict the test cases. Therefore, machine learning models such as Logistic Regression, Random Forest, and Decision Tree Classification’s, and SVM are all used for the implementation of our design to focus on the prediction of whether or not a person makes $50,000 a year with the census data through the use of various machine learning models and algorithms.

The paper consists of five sections excluding the introduction section. The “Material” section consists of the discussion of all the utilized models and algorithms implemented on the data for the project. The “Data” section containing the necessary information to understand the data we are working with. The “Feature Selection” column that discusses all of the important. The “Results” section that comprises the summarized result of our analysis in one location. Lastly, the “Conclusion” section warps up the analyzed result outcome of our project.

# Materials

In this project, we are performing a binary classification. To complete this task, we used a handful of different supervised learning models, and our own strategy for finding the best performance. Since our dataset conveniently comes with a set of true labels for every example, we were able to utilize supervised learning. We started off with a basic Logistic Regression model and eventually escalated into some more sophisticated models, such as a Random Forest. For our own strategy for finding the best performance, rather than taking the dataset we were given and preprocessing it from the beginning and using that until the end, we made several different versions of our dataset, using various preprocessing methods, and used them all throughout the whole project.

## Logistic Regression

Logistic Regression is a type of binary regression conducted when the dependent variable is dichotomous (binary) [12]. Similarly, to all regression analysis, a logistic regression is a predictive analysis that is utilized to describe the data presented and to further explain the coexisting relation between a binary dependent variable, and a binary independent variable.

## Random Forest Classifier

Random Forest consists of many independent decision trees that combine together to create an ensemble where each tree is dependent on the random vector’s sample value in an individual manner [3]. Random Forest divides the provided dataset features into a subset, where each individual tree spits out a class to build a classification tree, which in term produces our prediction. These classification trees can be used for training and testing purposes with the Random Forest taking in an estimator and a maximum feature value to later select the hyperparameter for the tree.

## Decision Tree Classifier

Decision Tree is a type of supervised machine learning algorithm where the data presented is constantly splitting into separate branches as the decision that was made changes accordingly to the set parameter [11]. A decision tree contains two entities, a decision node, and leaves. The node is explained as the splitting point where the data separates, whereas, the decision leaves are classified as the decided parameter, or final outcome [8].

## SVM

Support Vector Machine, or SVM is a type of supervised machine learning algorithm where the data features presented is plotted as a point in the N-dimensional space with each feature value being a specific coordinate related to the N-dimensional plotted space [9]. After plotting all the features given by the data, a classification can be done by finding the hyperplane that uniquely classifies the given data points into two different and separate classes [10]. An SVM is mostly used in classification problems, however, it can also be used for regression as well.

## StandardScaler

StandardScaler is a technique used to standardize the individual feature presented in the data inside of a specified range [1]. The StandardScaler technique is utilized by subtracting the mean and scaling to the unit variance meaning all the values must be divided by the standard deviation. The StandardSaler is performed during the pre-processing portion of the data analysis in order to handle the high variance of magnitudes, values, and units.

## SMOTE

Synthetic Minority Oversampling Technique or SMOTE for short is a statistical over-sampling technique where the number of cases increases in the dataset as synthetic samples of the existing minority cases are being generated, and then added into the provided dataset to balance the data in a proper fashion in order to help overcome the over-fitting problem that may be posed by the random oversampling [5].

## GridSearchCV

GridSearchCV is a model selection step derived from the library function that is the member of the sklearn’s model selection package [4]. The GridSearchCV assists with looping through all the predefined hyperparameters in order to fit the estimator on the training set as the best parameters from the listed hyperparameters can be selected at the end specifically tailored for the model. The GridSearchCV is performed after the data processing portion of the analysis.

# Data

## Dataset Description

The dataset was extracted from the census bureau database. There are 32,561 different instances, mix of continuous and discrete. We split this dataset into a train and test set with a 30% split size (train=22,800, test=9,761). The dataset has 15 attribute which include age, sex, education level and other relevant details of not only a person’s character, but more importantly, of their career statistics.

## Data Analysis

**Table 1** shows the description of all the attributes that exist within the dataset which includes the serial number, name, a small description of what the attribute is, as well as what data type the feature is categorized as.

|  |  |  |  |
| --- | --- | --- | --- |
| **S. no.** | **Attributes** | **Description** | **Data type** |
| 1 | age | Age (years) | Numerical |
| 2 | workclass | Work class | Categorical |
| 3 | fnlwgt | Final weight | Numerical |
| 4 | education | Education level | Categorical |
| 5 | education-num | Years of education | Numerical |
| 6 | marital-status | Marriage status | Categorical |
| 7 | occupation | Job Title | Categorical |
| 8 | relationship | Relationship to immediate family | Categorical |
| 9 | race | Race | Categorical |
| 10 | sex | Male/Female | Categorical |
| 11 | capital-gain | Amount of capital gain | Numerical |
| 12 | capital-loss | Amount of capital loss | Numerical |
| 13 | hours-per-week | Number of hrs. worked per week | Numerical |
| 14 | native-country | Country of origin | Categorical |
| 15 | income | Is income >50K or <50K? | Categorical |

**Table 1: Data Description Table**

# Feature Selections

From the dataset we were using (the Census Income dataset), there were 14 features available (not including the target column “income”) to work with, with an uneven split of 6 numerical features and 8 categorical. There were a few values missing from each column (unfortunately filled with a “?” instead of nothing, which made removing them ever so slightly harder), but there were nowhere near enough to justify removing any features, so we simply removed that row, which was a better decision overall. One feature that was a little more complicated to deal with was the “fnlwgt” (final weight) feature. This feature is a special custom type of feature, specific to this dataset. The weights correspond to “independent estimates of the civilian noninstitutional population of the US” [1]. In this single weight score is 3 “controls.” These controls are “a single cell estimate of the population 16+ for each state”, “controls for Hispanic Origin by age and sex”, and “controls by Race, age and sex” [1]. In our data analysis, we found this feature to essentially lump together three existing features already present in the dataset: age, race, and sex.

Each of these features in one way or another correlates with an individual having a high salary, so we felt it wasn’t appropriate to drop any of the columns. However, we were unsure of how to handle the “fnlwgt” feature. We were also curious about how this dataset would perform if we were to get rid of all qualitative data present. Due to this uncertainty, we simply decided on creating several versions of the dataset and testing them all. These versions include:

* Only numerical features
* All features present (one-hot encoded)
* Only the “fnlwgt” column dropped
* “Age”, “Race”, and “Sex” columns dropped
* Scaled data using the standard scaling method
* Rebalanced data using SMOTE

All these variations of the dataset would be tested first on our logistic regression model, then of these six, the best performing one would then be taken to be tested on our other three chosen models; random forest classifier, a decision tree classifier, and a support vector machine.

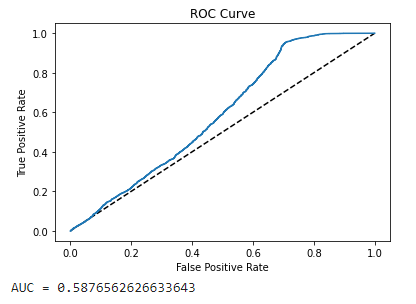
# Results

Each of the six dataset variations mentioned in **Section III** were split into training and test sets, with a 30% split. Each of our models were tested on the best performing variation (scaled data). For each of the models used, we printed a handful of performance metrics. A confusion matrix, a classification report, which contains precision, recall, F1 and accuracy scores, the ROC curve and AUC value, and lastly, the precision-recall curve. We felt that these performance metrics gave us the best impressions for the quality of our models.

The confusion matrix gives us the direct number of correctly classified and misclassified predictions, the most straightforward way of reading our model’s performance. It’s a good starting point, but it’s not enough. The classification report gives us a heap of extra useful information, which includes some more specific measures. The precision for accuracy based off the positive predictions made, the recall for the ratio of correctly classified positive instances, and the accuracy score for all correctly classified instances over the whole model. These metrics are able to give us the same results as the confusion matrix, but it makes it easier for us to read exactly how the model’s classifying everything. Using these values to create a precision-recall curve is also helpful as we can use the threshold values found to help us fully optimize both our precision and recall scores. Then we have our ROC curve and AUC score. AUC values close to 1 indicate good classifiers, so this was important for us to see.

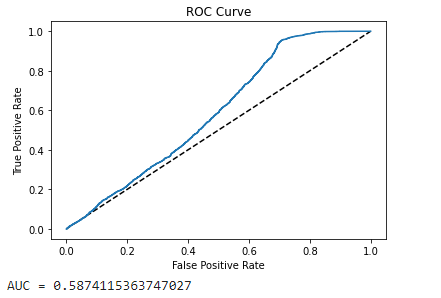
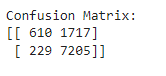
All figures shown below are each individual model’s performance. Reiterating **Section III**, the first half of which are the dataset variations, all using Logistic Regression. The last 3 figures refer to the other model types we chose using the best performing dataset. For each model, the confusion matrix, ROC curve is shown.

*Logistic Regression (numeric features)*



**Figure 1**

*Logistic Regression (one-hot encoding)*

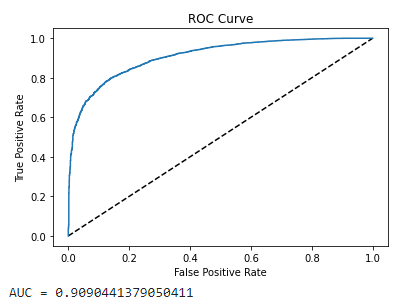
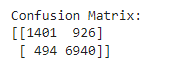


**Figure 2**

These first two figures show the most basic of models we could’ve made: using only numerical features and one-hot encoding categorical features to be numerical. They are also the models with the worst qualities, as one would expect. Not modifying the data in any way tends to give poor results. This is precisely why we chose to scale our data going forward.

Due to negligible improvements, the datasets of “fnlwgt” being the only column dropped, and dropping the “Age”, “Race” and “Sex” columns were omitted.

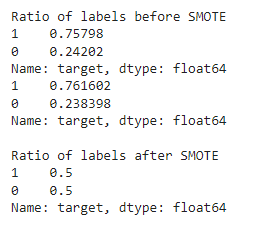
*Logistic Regression (scaled)*



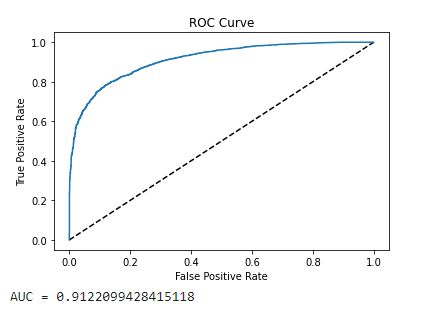
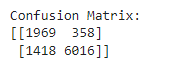
**Figure 3**

As you can see from **Figure 3**, scaling the data has drastically improved the performance of the Logistic Regression model, from an AUC of 58% shooting up to ~91% (not pictured, there was also a 5% increase in accuracy, from 80% to 85%). Scaling our data meant that the feature values were closer to each other overall rather than having just a few features dominate the rest and ultimately skew the data towards a worse performance. It also gives us slightly better overall execution time, which is always nice.

*Logistic Regression (SMOTE)*



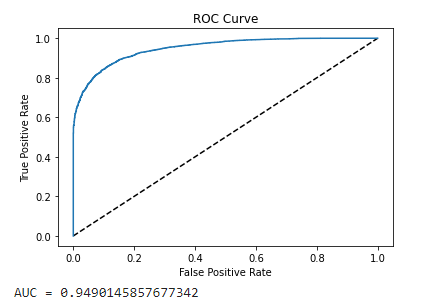
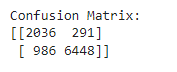
**Figure 4**



**Figure 5**

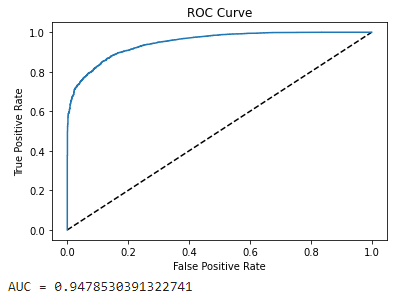
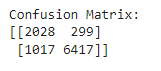
Our last variation involved rebalancing our dataset using SMOTE. In **Figure 4**, Looking at the ratio of the labels in the training (top) and test (bottom), they were leaning heavily towards 0 (made above $50K). Having an imbalanced dataset usually causes issues for some models, becoming biased towards predicting one label over the other. Using SMOTE on the dataset lets us even the playing field, so to speak, to try to avoid that bias. **Figure 5** shows that there was only a slight improvement (90.9% to 91.2%), but an improvement nonetheless. Content with these results, we used this dataset to predict the other models.

*Random Forest*



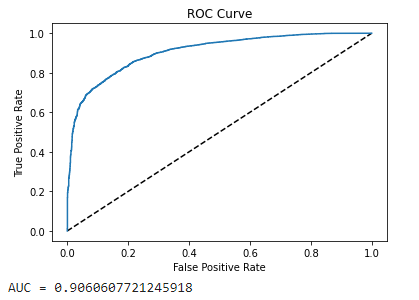
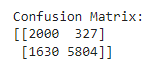
**Figure 6**

*Decision Tree*



**Figure 7**

*SVC*



**Figure 8**

For our other three models, we chose to use a RandomForestClassifier, a DecisionTreeClassifier, and an SVC support vector machine. We chose these three specifically because Random Forest is one of the most powerful algorithms we have in our arsenals, Random Forests are made up of several Decision Trees so we wanted to see how a single one would perform, and SVC is typically sensitive to scaled data, so we wanted to see how it would fare with our current dataset.

As you would expect, the SVC (**Figure 8**) performed the worst out of the three, even coming slightly behind the normal Logistic Regression model. This isn’t very surprising considering SVC and support vector machines in general are typically sensitive to scaled data.

Random Forest and Decision Tree however, managed to bring our AUC value up a whole ~3.6% compared to Logistic Regression. Decision Tree surprisingly won out over Random Forest, but only barely, scoring 0.02% higher.

# Conclusion

The census data collects information about individual’s race, families and living arrangements, as well as a person’s health, marital status, education, employment, and housing, etc. An individual’s economic measure, determining whether a person’s income is either low or high according to the census data can be predicted by the provided features.

Predicting a person’s income using the census data can be beneficial to many individuals. Specifically in a real-life occurrence where those individuals who may have been determined to have a low income can qualify to participate in government funded programs such as Medicaid, and Food Stamps which can help those individuals in severe need of financial assistance who are not fortunate enough to afford those services.

With that in mind, using all the baseline and inbuilt machine learning models used to predict a person’s income, we can confidently conclude that we were able to make a correct prediction on the given dataset.

While our first two dataset variants and models gave us less than stellar results, we were very satisfied with our results after we did some preprocessing. The AUC scores before scaling the data were barely reaching 60%, but afterwards, they were all 90% or above, with the highest peaking at 94.8%, which indicates a *very* goodmodel. A primary goal of ours was simply being able to predict if a person makes an income above a certain threshold, but we ended up achieving a secondary goal as well, that being showing how much preprocessing your data can affect the quality of a model.

Overall, the project was a success in meeting our overall objective goals in creating this predictive model, as well as meeting our desired result specifications.

##### Acknowledgments

This project could not have been completed without the assistance and guidance from our professor and instructor Prof. Ehsan Kazemi Foroushani for his informative guidance and direct instructions that he has given us throughout this entire process by providing us with feedback to our project every week to ensure that we were on the right track in completing this project correctly.

We would also like to thank our Teaching Assistant Ms. Madeline Schiappa for taking the time out of her day to help us with our project whenever we got stuck at a certain step of development for our project, and for overall helping us when we needed help.

##### References

1. A. Kharwal, “Standardscaler in machine learning,” Data Science | Machine Learning | Python | C++ | Coding | Programming | JavaScript, 24-Jun-2021. [Online]. Available: https://thecleverprogrammer.com/2020/09/22/standardscaler-in-machine-learning/#:~:text=In%20Machine%20Learning%2C%20StandardScaler%20is,the%20standard%20deviation%20is%201. [Accessed: 26-Apr-2022].
2. “Census income · master · data science dojo / datasets,” Code. [Online].Available: https://code.datasciencedojo.com/datasciencedojo/datasets/tree/master/Census%20Income. [Accessed: 26-Apr-2022].
3. “Decision trees for classification: A machine learning algorithm,” Xoriant, 07-Sep-1970. [Online]. Available: https://www.xoriant.com/blog/product-engineering/decision-trees-machine-learning-algorithm.html#:~:text=Introduction%20Decision%20Trees%20are%20a,namely%20decision%20nodes%20and%20leaves. [Accessed: 26-Apr-2022].
4. “GRIDSEARCHCV: Tune hyperparameters with GRIDSEARCHCV,” Analytics Vidhya, 15-Jul-2021. [Online]. Available: https://www.analyticsvidhya.com/blog/2021/06/tune-hyperparameters-with-gridsearchcv/#:~:text=GridSearchCV%20is%20a%20model%20selection,of%20Tuned%20and%20Untuned%20Models. [Accessed: 26-Apr-2022].
5. Likebupt, “Smote - Azure Machine Learning,” Azure Machine Learning | Microsoft Docs. [Online]. Available: https://docs.microsoft.com/en-us/azure/machine-learning/component-reference/smote#:~:text=Synthetic%20Minority%20Oversampling%20Technique%20(SMOTE,that%20you%20supply%20as%20input. [Accessed: 26-Apr-2022].
6. N. Jain, S. Jhunthra, H. Garg, V. Gupta, S. Mohan, A. Ahmadian, S. Salahshour, and M. Ferrara, “Prediction modelling of COVID using machine learning methods from B-cell dataset,” Results in Physics, 17-Jan-2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2211379721000012#f0080. [Accessed: 26-Apr-2022].
7. O. E. C. D. S. Directorate, OECD Glossary of statistical terms - census definition. [Online]. Available: https://stats.oecd.org/glossary/detail.asp?ID=301. [Accessed: 26-Apr-2022].
8. P. Gupta, “Decision trees in machine learning,” Medium, 12-Nov-2017. [Online]. Available: https://towardsdatascience.com/decision-trees-in-machine-learning-641b9c4e8052. [Accessed: 26-Apr-2022].
9. R. Gandhi, “Support Vector Machine - introduction to machine learning algorithms,” Medium, 05-Jul-2018. [Online]. Available: https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47. [Accessed: 26-Apr-2022].
10. “SVM: Support Vector Machine Algorithm in machine learning,” Analytics Vidhya, 26-Aug-2021. [Online]. Available: https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/. [Accessed: 26-Apr-2022].
11. “What is a decision tree?,” Master's in Data Science. [Online]. Available: https://www.mastersindatascience.org/learning/introduction-to-machine-learning-algorithms/decision-tree/. [Accessed: 26-Apr-2022].
12. “What is logistic regression?,” Statistics Solutions, 11-Aug-2021. [Online]. Available: https://www.statisticssolutions.com/free-resources/directory-of-statistical-analyses/what-is-logistic-regression/. [Accessed: 26-Apr-2022].
13. “What is the average American income in 2021?,” PolicyAdvice, 12-Mar-2022. [Online]. Available: https://policyadvice.net/insurance/insights/average-american-income/#:~:text=The%20average%20annual%20wage%20in,average%20and%20median%20wage%20data. [Accessed: 26-Apr-2022].