**Machine Learning Classification Project: Finding Annual Income (Python)**



The recent coronavirus outbreak has seen a tremendous amount of people who signed up for the stimulus checks of $1200 in America after losing their jobs. One of the requisites for the recipients is to have their annual incomes less than $75,000, which is all accessible through their annual tax report. (Singletary , 2020) Nonetheless, there is a staggering amount of people who have not done their taxes. Understanding the potential annual income of unfiled taxes individuals can help the government to make strategic steps in taking care of them and ensuring the size of their bank reserve. Thus, the government would benefit tremendously from a machine learning model that can help predict an individual's income base on their demographic features.

In this project, we will use five different supervised algorithms (KNN, Naïve Bayes, Decision Tree and Rules, Random Forests, GLM & Logistic Regression.) on the dataset from the 1994 US Census. We will select the best candidate algorithms when comparing the preliminary results with each other and further optimize the chosen algorithm to best model the data. Our goal with this project is to finally come up with the model that can accurately predict whether an individual earns more or less than $50,000 (Classification- binary class). According to the Inflationcalcualtor.com, $50,000 in 1994 equals to $87,000 in 2019. Then after that, future researchers can use this model to apply for the current year's dataset.

The dataset for this project comes can be accessed from the UCI Machine Learning Repository or Kaggle website. The dataset is donated thanks to Ron Koshavi and Barry Becker after they published their findings in "Scaling Up the Accuracy of Naive-Bayes Classifiers: A Decision-Tree Hybrid" (Kohavi, 1996). Bearing in mind the fact that the data we are studying here consists of small changes to the original dataset like removing the 'fnlwgt' feature and records with missing or ill-formatted entries.

# Data

The changed census dataset in this project has approximately 32,000 entry points, with each entry points having 13 features.

**Features**

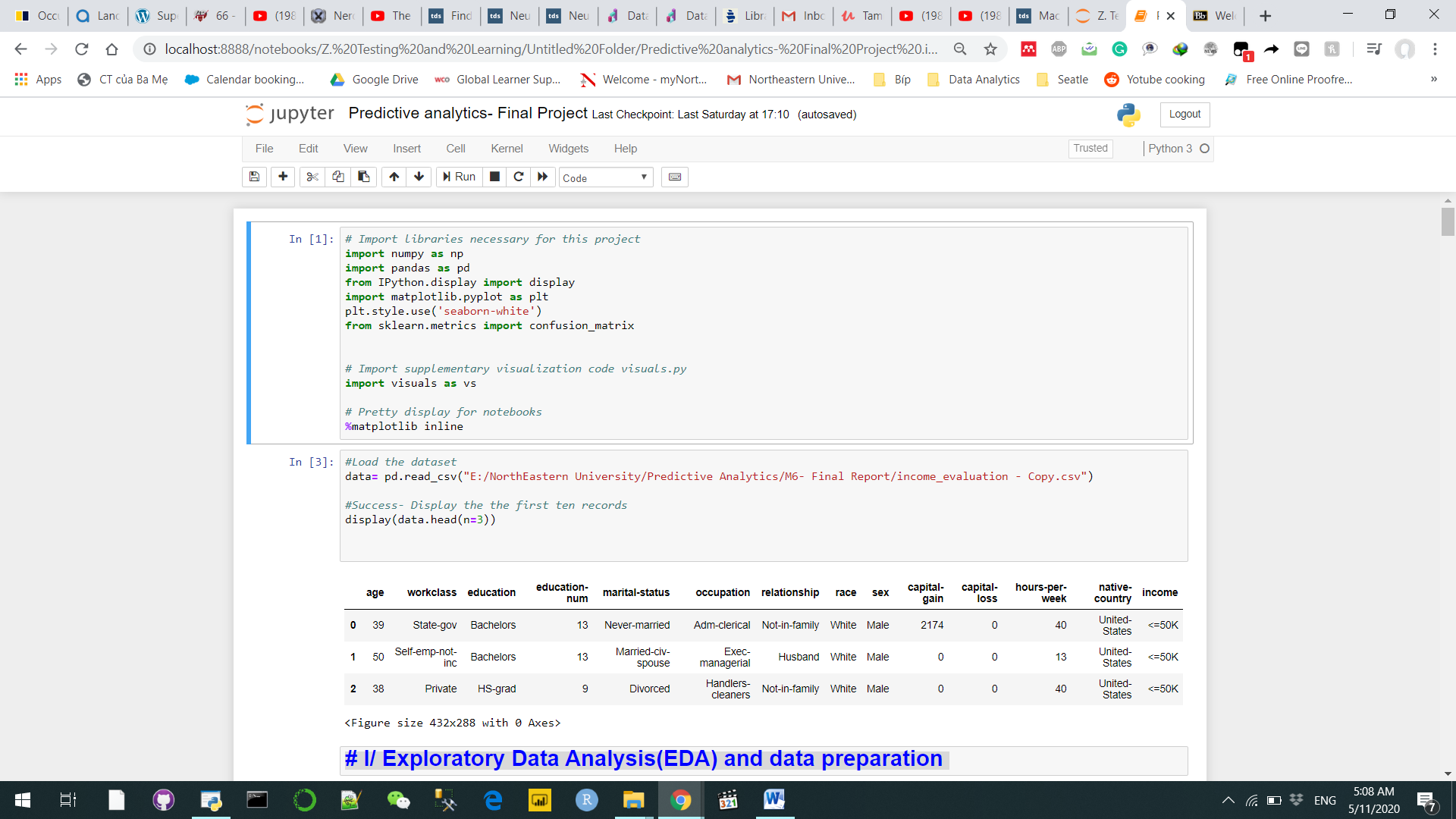
* age: Age
* workclass: Working Class (Private, Self-emp-not-inc, Self-emp-inc, Federal-gov, Local-gov, State-gov, Without-pay, Never-worked)
* education\_level: Level of Education (Bachelors, Some-college, 11th, HS-grad, Prof-school, Assoc-acdm, Assoc-voc, 9th, 7th-8th, 12th, Masters, 1st-4th, 10th, Doctorate, 5th-6th, Preschool)
* education-num: Number of educational years completed
* marital-status: Marital status (Married-civ-spouse, Divorced, Never-married, Separated, Widowed, Married-spouse-absent, Married-AF-spouse)
* occupation: Work Occupation (Tech-support, Craft-repair, Other-service, Sales, Exec-managerial, Prof-specialty, Handlers-cleaners, Machine-op-inspct, Adm-clerical, Farming-fishing, Transport-moving, Priv-house-serv, Protective-serv, Armed-Forces)
* relationship: Relationship Status (Wife, Own-child, Husband, Not-in-family, Other-relative, Unmarried)
* race: Race (White, Asian-Pac-Islander, Amer-Indian-Eskimo, Other, Black)
* sex: Sex (Female, Male)
* capital-gain: Monetary Capital Gains
* capital-loss: Monetary Capital Losses
* hours-per-week: Average Hours Per Week Worked
* native-country: Native Country (United-States, Cambodia, England, Puerto-Rico, Canada, Germany, Outlying-US(Guam-USVI-etc), India, Japan, Greece, South, China, Cuba, Iran, Honduras, Philippines, Italy, Poland, Jamaica, Vietnam, Mexico, Portugal, Ireland, France, Dominican-Republic, Laos, Ecuador, Taiwan, Haiti, Columbia, Hungary, Guatemala, Nicaragua, Scotland, Thailand, Yugoslavia, El-Salvador, Trinadad&Tobago, Peru, Hong, Holand-Netherlands)

**Target Variable**

* income: Income Class (<=50K, >50K)

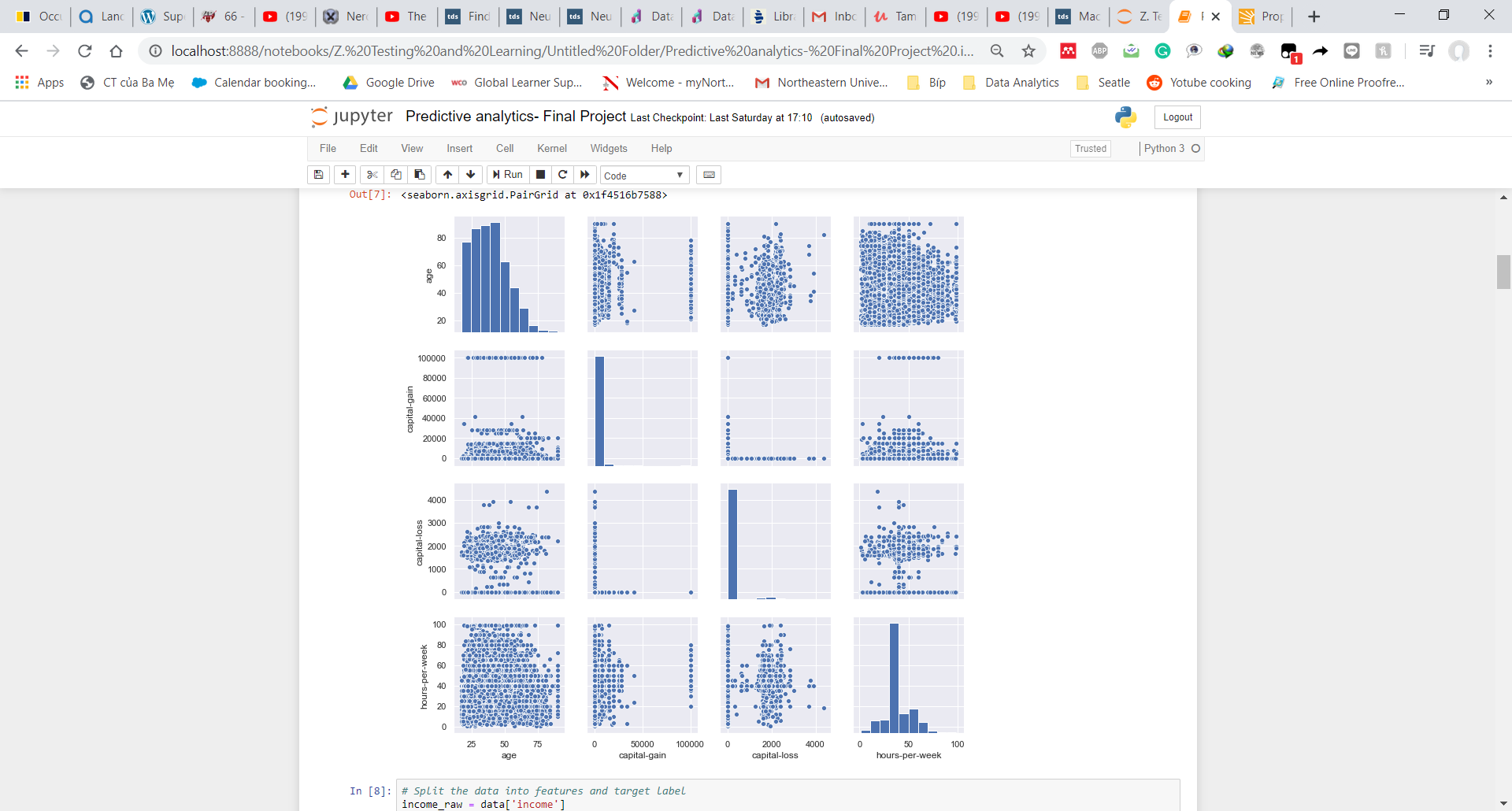
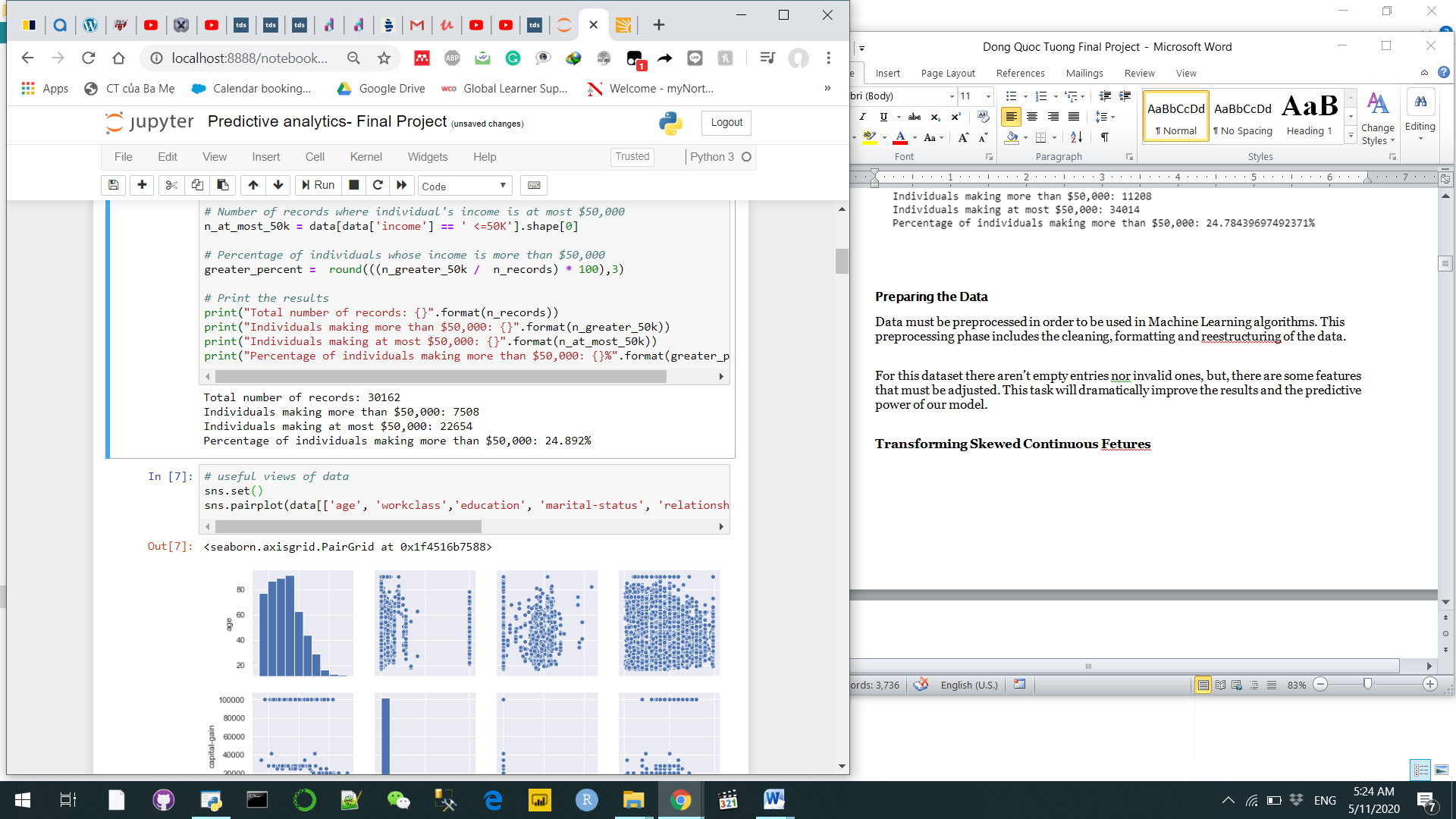
# Import Libraries and Load Data

We will first import the dataset as well as the necessary Python libraries that are useful for our analysis like numpy, pandas, IPython, matplolib, seaborn, visuals file. The last column of the dataset is the final 'income' binary outcome and the rest are features



**I/ Exploratory Data Analysis(EDA) and data preparation**

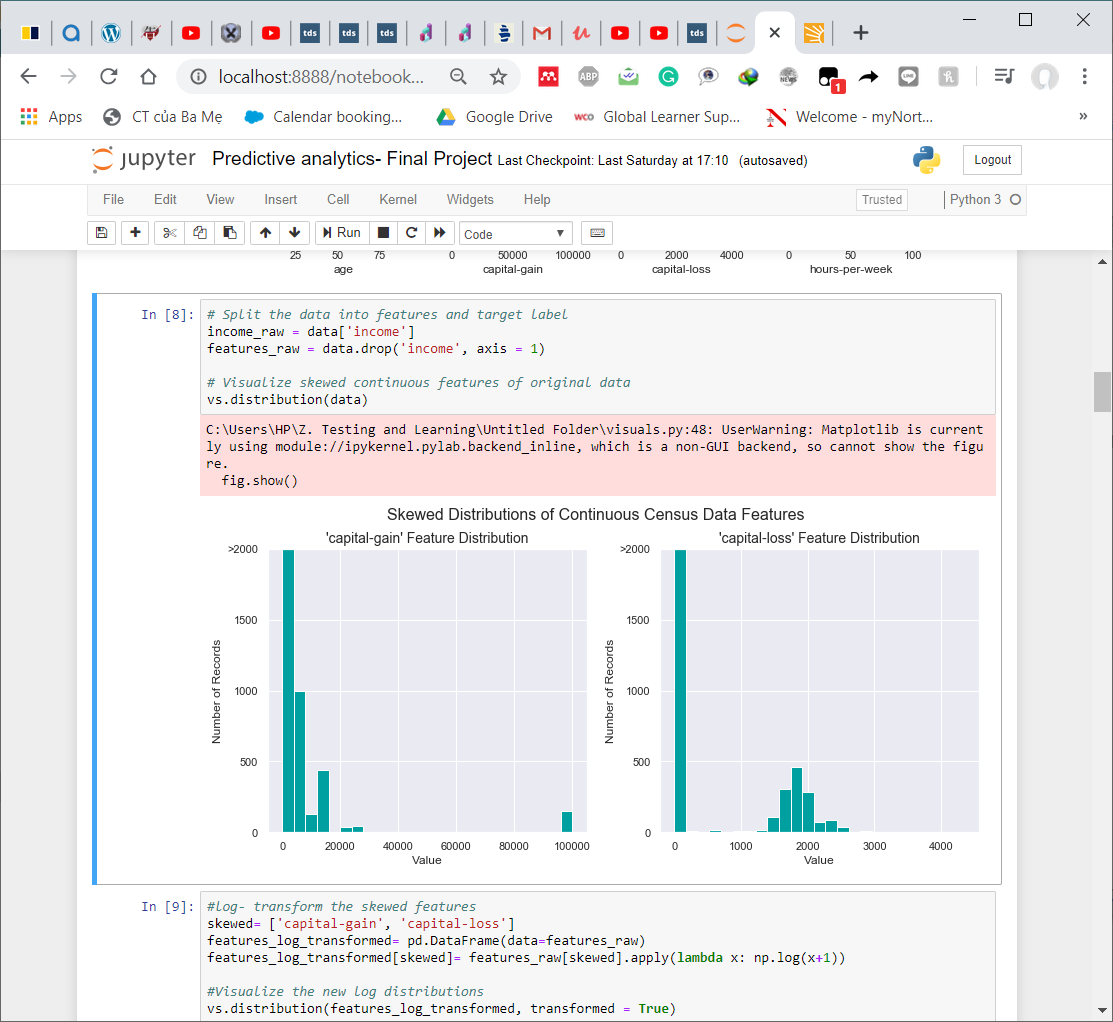
We explore our dataset a little bit to see how many individuals fit in each group and the percentage of citizens that earn more than $50,000 annually. We can also have the visual views of dataset to see how each factor interact with each other through seasborn.pairplot() function



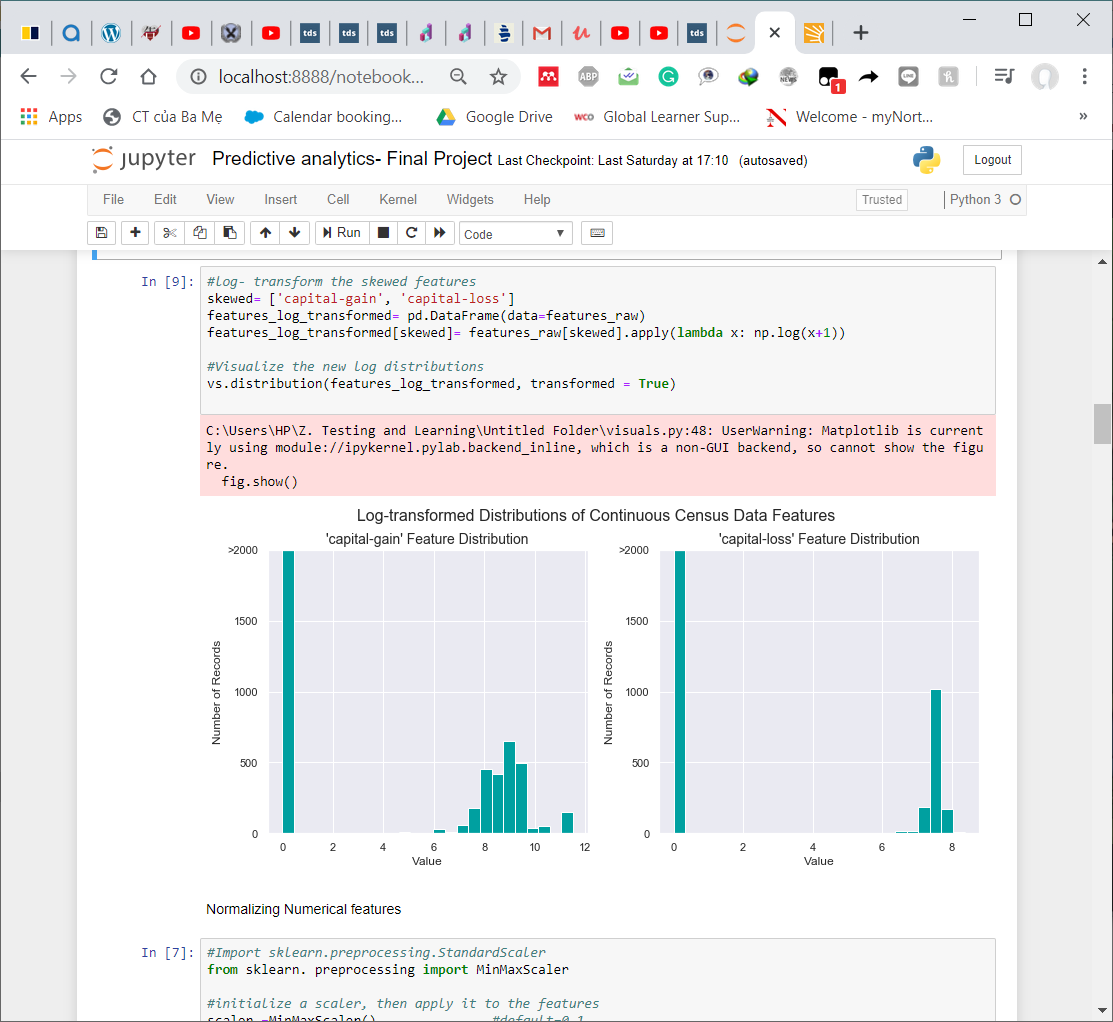
Data must be prepared in order to be accepted in the Machine Learning algorithms. If not, the model will make incorrect predictions. This process includes cleansing, formatting, and feature scaling of the data.

1/ The first step is to transform skewed continuous features. It is due to the fact that skewed distributions on the features' values can make an algorithm to underperform if the range is not normalized. We split the data into features and target labels. Then we visualized skewed continuous of original data. The income dataset has two features with skewed distribution:

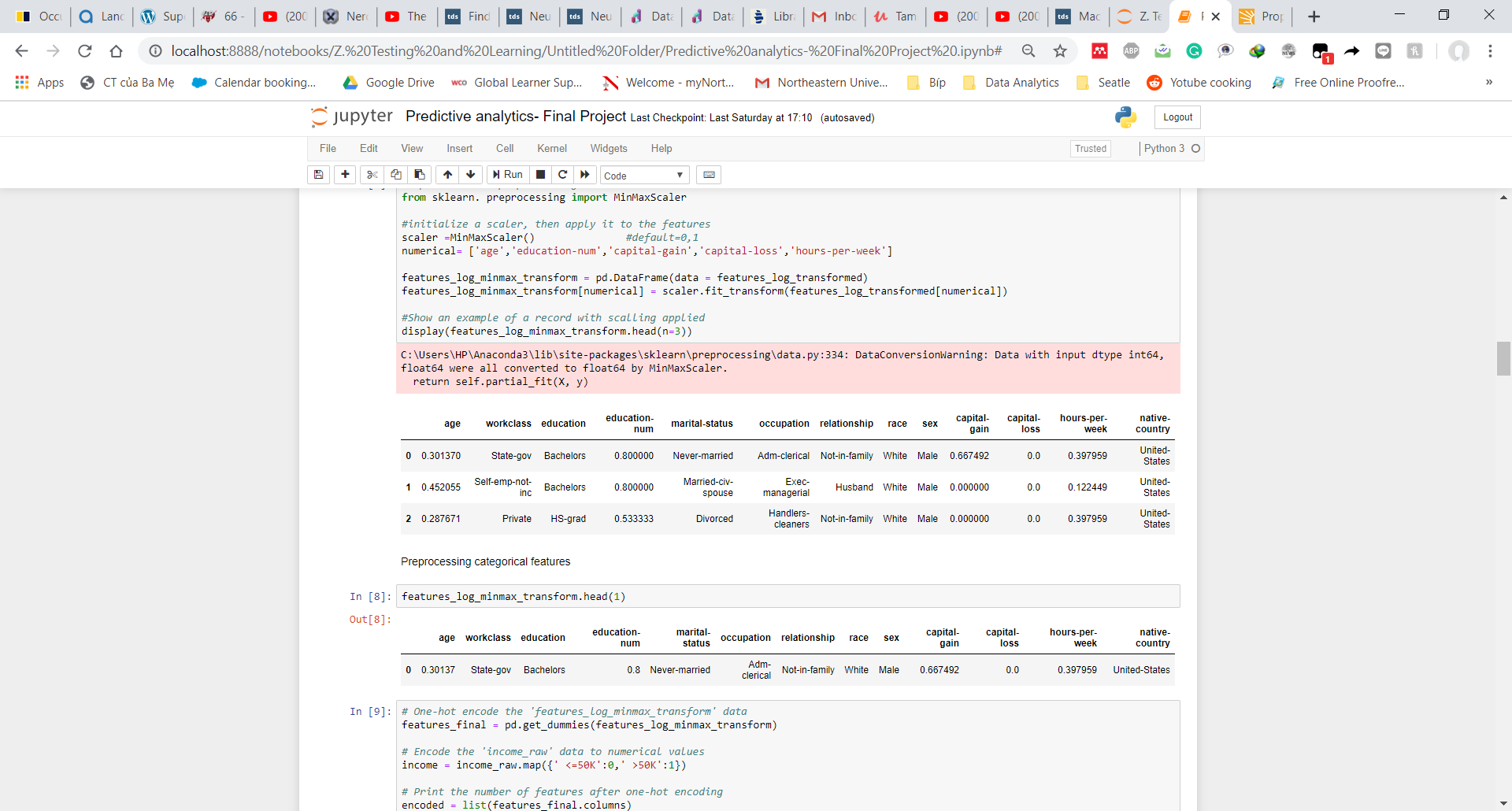
* ‘capital-gain’
* ‘capital-loss’



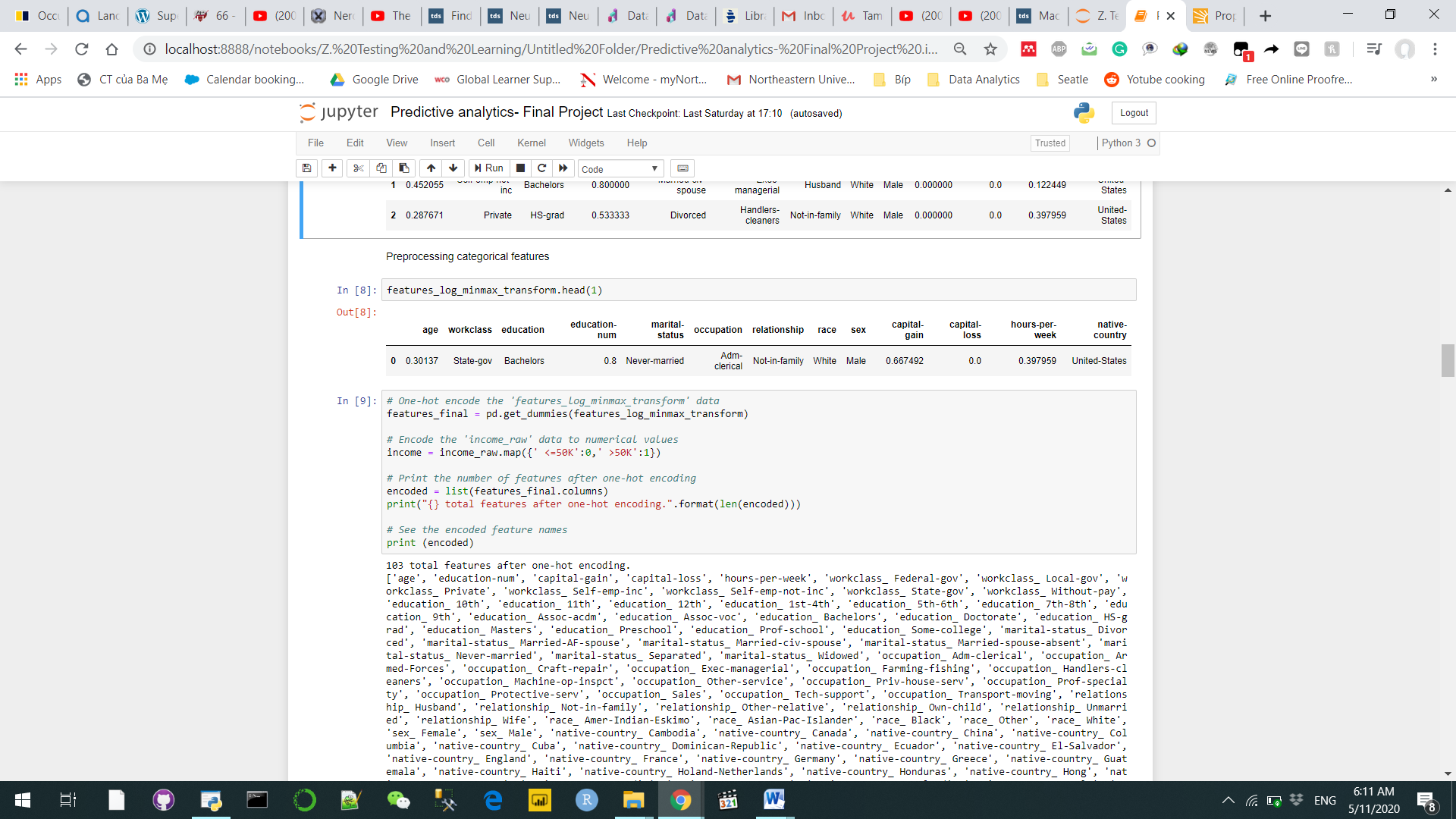
The features' distribution skewed to the right. Therefore, we apply the logarithmic transformation on the data to prevent outliners from having negative impacts on the machine learning model later on. Nevertheless, one must be careful in treating the o values since the log(o) is undefined. We can translate these values into small amounts above o so as to apply the logarithm well.



2/ The second step is to normalize numerical features through data scaling. This scaling will not ultimately change the shape of the distributions of the features but it makes sure that there is an equal treatment to every single feature when the five models are applied.

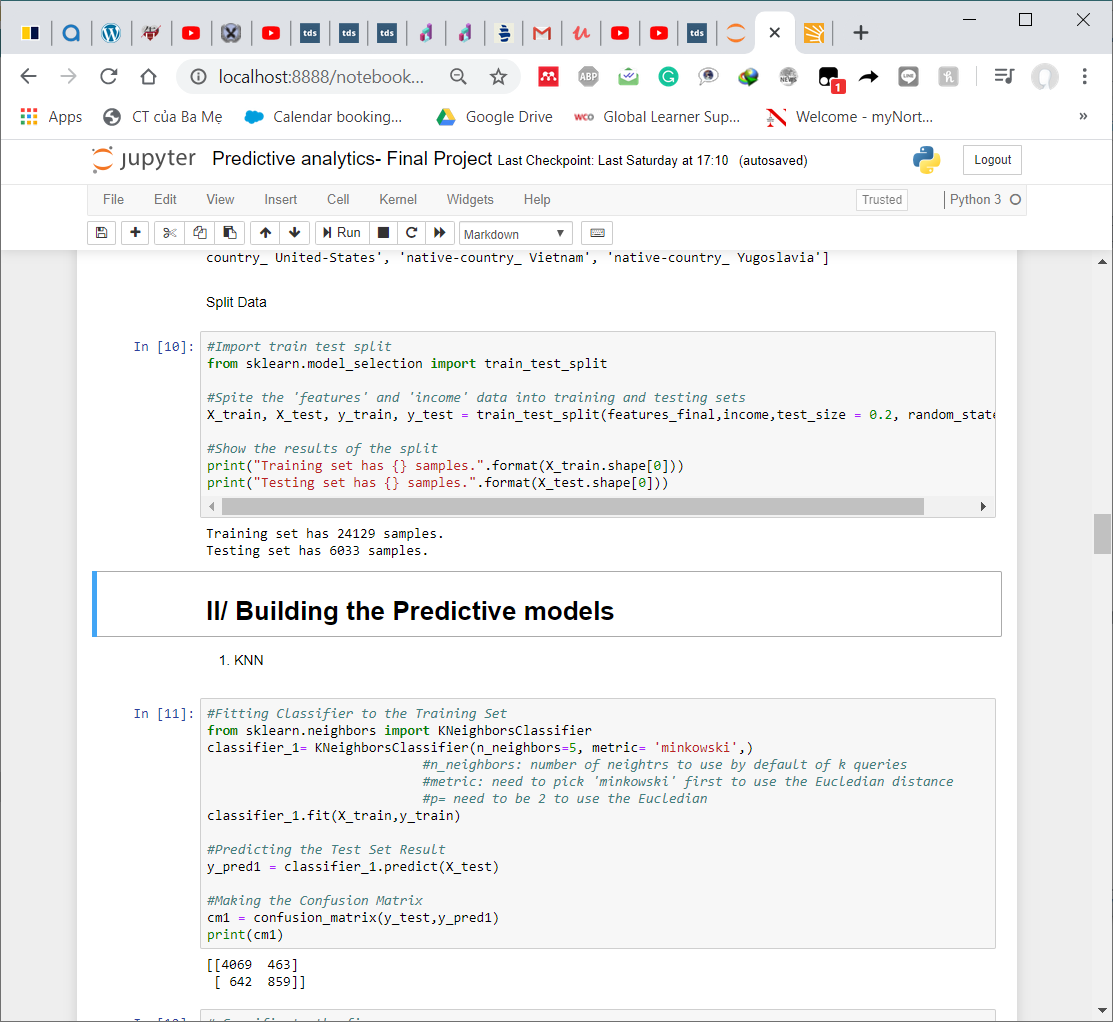


**3/ The third step is to turn categorical factors like ‘’occupation’ or ‘race ‘into numerical ones using the one-hot encoding function. By doing so, the algorithm will create ‘dummy’ variable for each possible category of the categorical features which are similar to the numerical features. However, one-hot-encoder is not suitable for the target suitable because it is a complicated procedure, thus we will manually assign ‘<=50k ‘ as 0 and ’>50k’ as 1 to avoid unexpected fallout at the end. From 13 unique features, the dataset was tuned into 103 total features thanks to one-hot encoding function.**





**4/ The fourth step is to shuffle and split dataset with 80-20 ratio for training the testing, respectively**



**II/ Building the Predictive models**

# Considering the shape of our data (30162 data points with 103 unique factors), we are fortunate enough that all the models that we choose can handle the significant amount of factors appropriately. To access each model correctly, we will compare each model's strengths and weaknesses

**a) K-Nearest Neighbor Algorithm**

* **Strengths**: It is simple with no training period (known as Lazy Learner). Researchers can add new data seamlessly without affecting the accuracy of the algorithm
* **Weaknesses**: KNN is incompatible with datasets that are large or have high dimensions. The model also needs feature scaling and is highly sensitive to noisy data, missing values as well as outliners. (Kumar, 2019)
* **Potential Application:** search applications for ‘’similar’ items

**b) Gaussian Naive Bayes**

* **Strengths**: One of the easiest and fastest classifiers that provide good results with little tunning of the model’s hyperparameters. Additionally, it does not require a large amount of data to be effectively trained.
* **Weaknesses**: The model makes a greatly strong assumption about the shape of data distribution. It also suffers from data scarcity and continuous problems which can render the entire prediction inaccurate. Generally advised against using it for Classification problem
* **Potential Application:** natural language processing (text learning)

**c) Decision Tree**

* **Strengths**: The process has understandable rules and can perform classification without much computation. It can handle both continuous and categorical variables and provides the researchers with a clear indication of which features are important. (Geeksforgeeks, 2019)
* **Weaknesses**: Continuous attributes might not be most suitable for the model. It can prone to errors with many classes and a relatively small number of training examples and is quite costly to train.
* **Potential Application:** image classification for animals

**d) Random Forests**

* **Strengths**: Very suitable for the Classification problem because it is an ensemble of decision trees. Random Forest is also suitable with high dimensional spaces and a large number of training examples
* **Weaknesses**: it may overfit when dealing with noisy data
* **Potential Application:** stock market price prediction

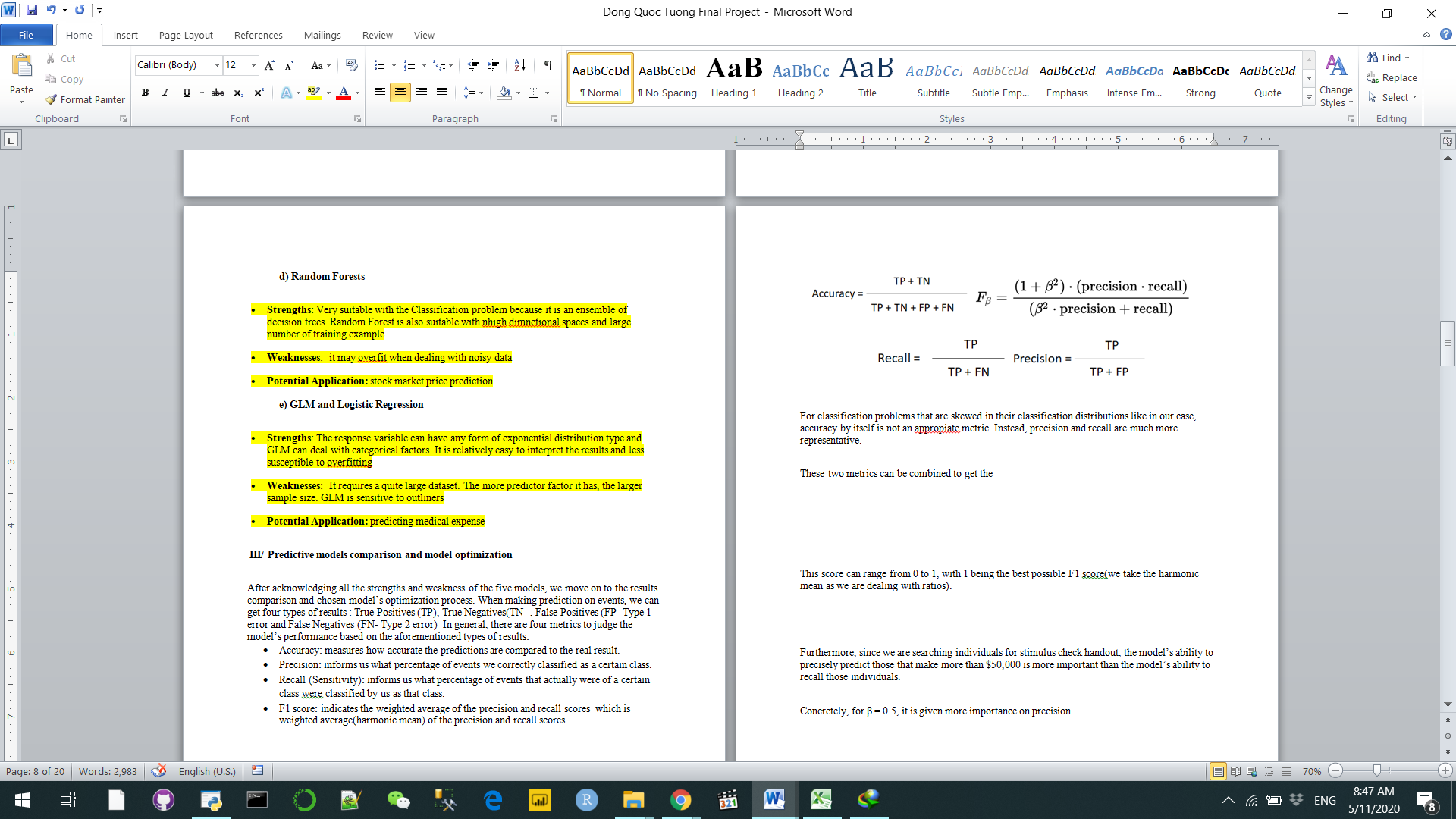
**e) GLM and Logistic Regression**

* **Strengths**: The response variable can have any form of exponential distribution type and GLM can deal with categorical factors. It is relatively easy to interpret the results and less susceptible to overfitting
* **Weaknesses**: It requires a quite large dataset. The more predictor factor it has, the larger the sample size. GLM is sensitive to outliners
* **Potential Application:** predicting medical expense

**III/ Predictive models comparison and model optimization**

**a) Models comparison**

After acknowledging all the strengths and weaknesses of the five models, we move on to the results comparison and chosen model's optimization process. When making a prediction on events, we can get four types of results: True Positives (TP), True Negatives(TN), False Positives (FP- Type 1 error) and False Negatives (FN- Type 2 error) In general, there are four metrics to judge the model's performance based on the aforementioned types of results:

* Accuracy: measures how accurate the predictions are compared to the real result.
* Precision: informs us what percentage of events we correctly classified as a certain class.
* Recall (Sensitivity): informs us what percentage of events that actually were of a certain class were classified by us as that class.
* F1 score: indicates the weighted average of the precision and recall scores which is the weighted average(harmonic mean) of the precision and recall scores

Due to the fact that our dataset is a classification problem that has some skewed distributions in the factors, accuracy itself is not an appropriate metric. Instead, we will look for precision and recall figures. Furthermore, since we are searching individuals that deserved stimulus check handout, the model’s ability to precisely predict those that make more than $50,000 is more important than the model’s ability to recall those individuals.

# 

# 

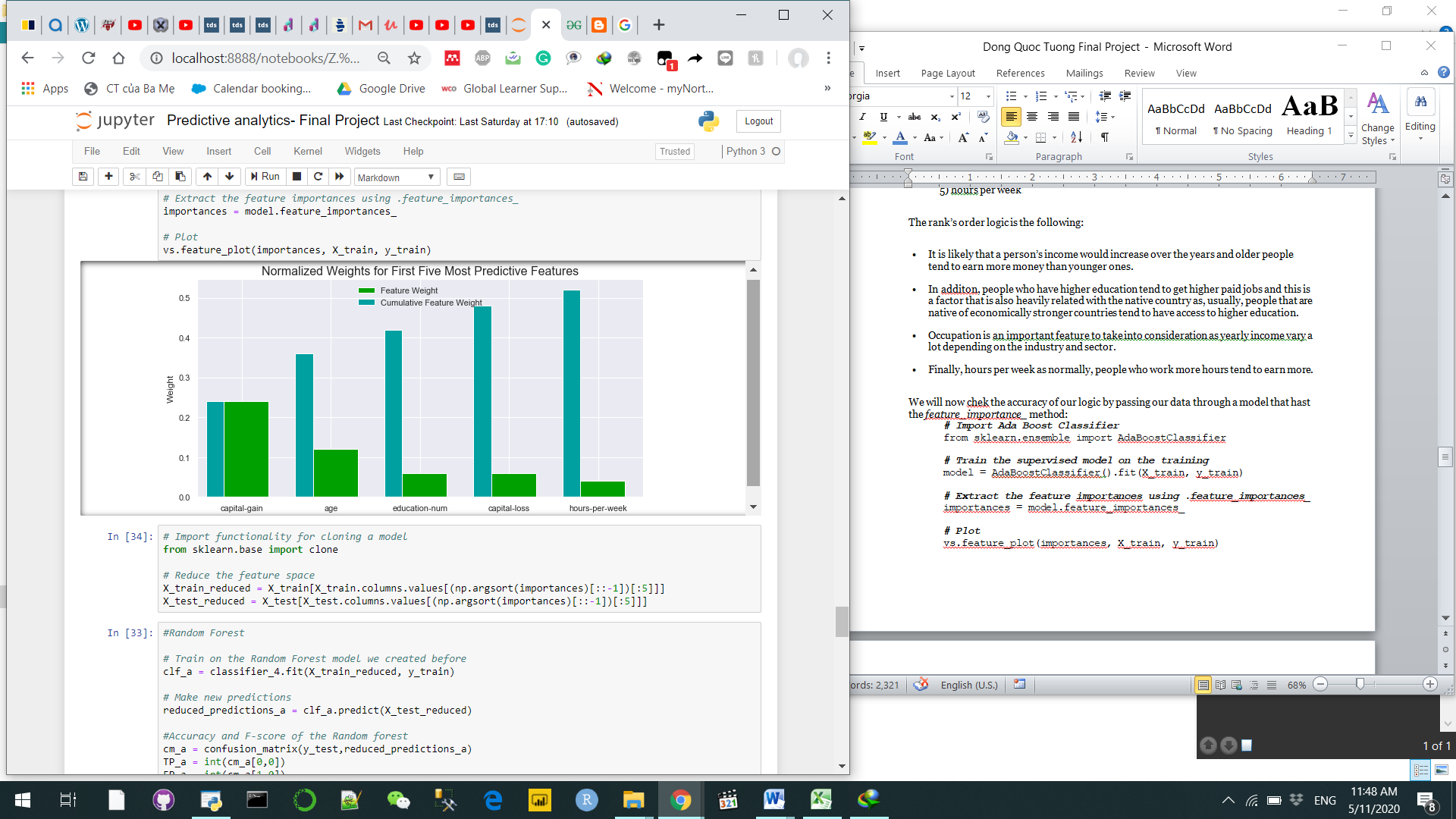
According to the table and graph above, Random Forest and Logistic Regression have the greatest precision and recall scores. Despite having a higher precision score, Random Forest 's difference from Logistic Regression's is very minor the difference is minor between Random Forest and Logistic Regression. It would be much better for us to improve the models and compare one more time again before making the final decision.

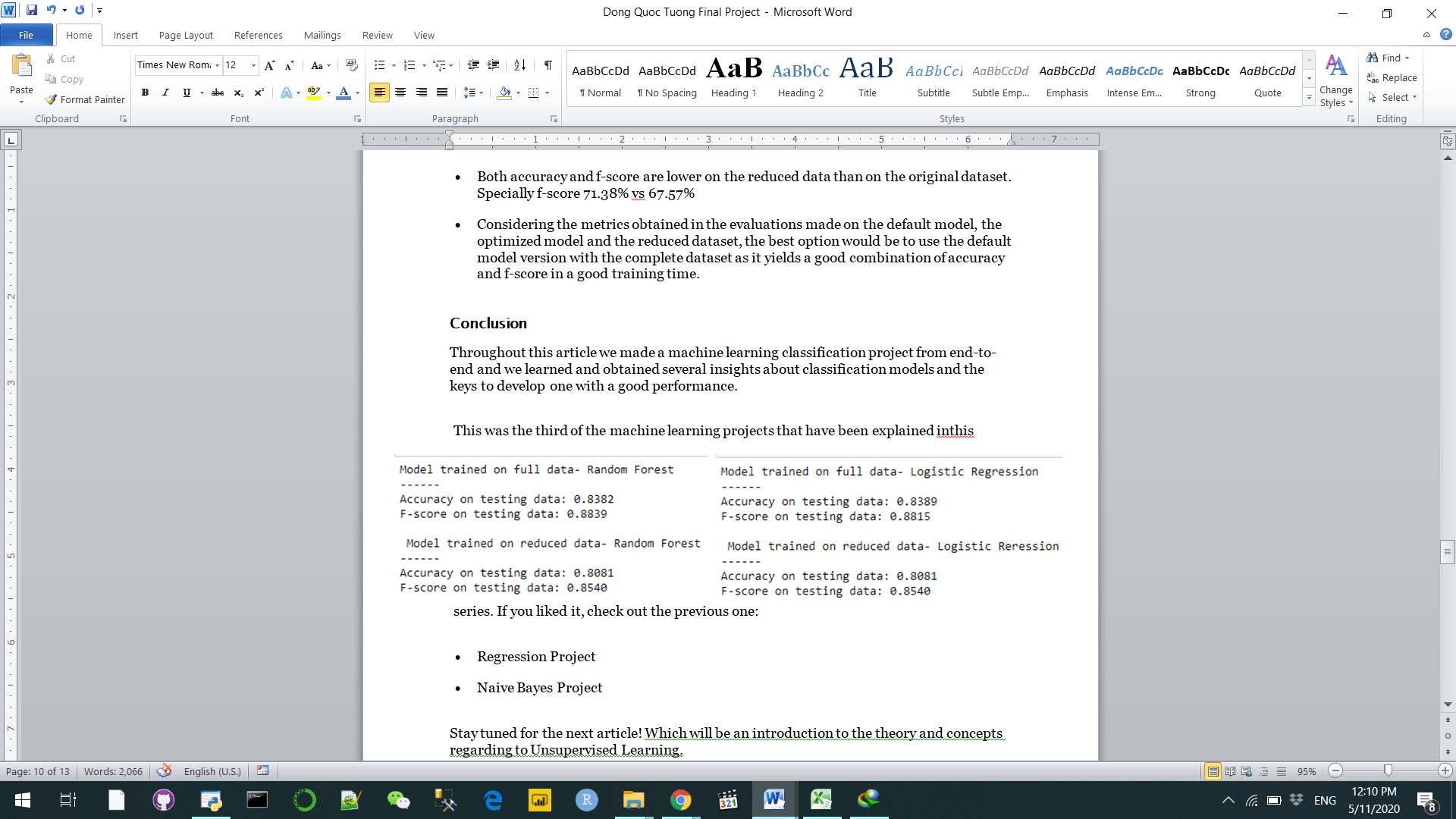
We will try to have how feature selections affect the performances of these two models and apply the K-fold Cross-Validation with different parameters and Grid search to find the best model with the best parameters.

**b) Models optimization**

**Features selection**

An important task when performing supervised learning on a dataset like this one is investigating the degree of impact the factors have on the final predictive performance. By distilling the relationships to only between only a few crucial features and the target variable, we can subsequently eliminate the noise. There are 13 features available on the original dataset and we want to cut them down to half. Intuitively, we will select factors like age, education, native country, occupations, and work hours per week because they are usually the best indicators of an individual's financial situation. Then, we check the accuracy of our logic by using the AdaBoostClassifier classifier to plot the graph below



Our intuition is partially accurate since age; hours per week and education leave a tremendous impact on the financial situation of a working person. Surprisingly, we failed to identify the importance of the capital gain and capital loss.

After re-performing the Random Forest and Logistic Regression, we witnessed a fall in both accuracy and F-score on the reduced data than the original dataset across two Predictive models. Considering the metrics obtained in the evaluations of the original model on the reduced datasets, the best option would be to keep using the original dataset versions as they yield a good combination of accuracy and f-score.

**K-Fold Cross Validation and Grid search**

# In this step, we used the Cross validation (rotation estimation) method to validate the model’s techniques by splitting the dataset into many folds and cross test the result on it. (Brownlee, 2019) The ‘estimator’ parameter indicates the object that you used to fit the model (Classifier for Random Forest and Logistic Regression) while ‘cv’ verified how many folds you want to split the training set into (usually 10). Then we compute the mean and standard deviations of 10 different accuracies based on 10 small dataset samples and compare it with each other.

# 

# 

# The Mean and Standard deviation of these two models after Cross Validation are identical to each other. It is safe to say that whichever model we pick between these two will not make much of a difference. Thus, we can perform the Grid Search solely on the Random Forest Model to choose the best parameters that could yield the highest possible accuracies. ‘n\_estimators’ has the options between 10, 30, 100, 300 while ‘criterion’ has the options between ‘entropy’ and ‘gini’ model

# 

# Finally, the most optimal model that could yield 0.844 accuracy score is Random Forest with ‘criterion’ = ‘entropy’, ‘n\_estimators’ =’300’

**IV/ Conclusion**

This project allowed us to not only have accurate predictions about the annual earning for each individual but also to understand each model's strengths and weaknesses when it comes to classification. Future researchers are advised to further improve the models and adapt them accordingly to fit the requirement of the problem

**Reference & Sources:**

Brownlee, J. (2019). A Gentle Introduction to k-fold Cross-Validation. Retrieved from <https://machinelearningmastery.com/k-fold-cross-validation/>

Geeksforgeeks. (2019, April 17). Decision Tree. Retrieved from <https://www.geeksforgeeks.org/decision-tree/>

Kohavi  , R. (1996). Scaling up the accuracy of Naive-Bayes classifiers: a decision-tree hybrid. *KDD’96: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, *8*, 202–207. Retrieved from <https://www.aaai.org/Papers/KDD/1996/KDD96-033.pd>

Kumar, N. (2019). Advantages and Disadvantages of KNN Algorithm in Machine Learning. Retrieved from <http://theprofessionalspoint.blogspot.com/2019/02/advantages-and-disadvantages-of-knn.html>

Singletary , M. (2020). Still waiting for your stimulus check? You have until 12 p.m. Wednesday to give the IRS your bank information. Retrieved from <https://www.washingtonpost.com/gdpr-consent/?next_url=https%3a%2f%2fwww.washingtonpost.com%2fbusiness%2f2020%2f05%2f11%2fstill-waiting-your-stimulus-check-you-have-until-12-pm-wednesday-give-irs-your-bank-information%2f>

UCI Machine Learning repository <https://archive.ics.uci.edu/ml/datasets/Census+Income>