Income Prediction based on Machine Learning Techniques

Borga Edionse Usifo
Indiana University
Bloomington, Indiana 47408
busifo@iu.edu

ABSTRACT

This project takes a closer look to some of the most used supervised learning algorithms in machine learning. We start with the description of the each of the algorithms then we move it to analytics and findings by using that particular algorithm in our data-set. We also provide advantages and disadvantages of each supervised machine learning algorithm for future reference. We mainly focus on our prediction of the income level of individuals by looking at their age, gender, education, location, and other features given by our data-set. We will try each algorithm and try to pick the best features from our data-set to have an optimal prediction.

KEYWORDS

i523, HID343, Machine Learning, Income Prediction, Logistic Regression, Ensemble methods

1 INTRODUCTION

In this project, we try to showcase the performance of the machine learning algorithms on data which we gather from UCI machine learning repository [22]. This data used by Kohavi R. and Becker B. for their research in improving the in Naive Bayes Classifier's accuracy [21].

Data consists of 15 variables, and we try to predict the income of the individuals. To do this prediction task, we first started with data preparation because the data we receive from UCI machine learning repository [22] not fully prepared for any machine learning algorithm. Our first task was the clean the data while applying some statistical techniques to get insights from the dataset. We also used data transformation methods like One-Hot-Encoding[45] to apply logarithmic functions for improving the machine learning algorithms performance before training the data.

Machine Learning algorithms that we discuss in this paper are Gaussian Naive Bayes [46], K Nearest Neighbors [29], Ensemble Methods (Boosting) [8], Support Vector Machines [6], Logistic Regression [34], and Decision Trees [49]. We try to show their weakness, advantages, and their time consumption while training each of them in machine learning algorithms section.

After providing a brief introduction of each of the supervised machine learning algorithms, we will discuss our findings for of each of the algorithms by comparing their accuracy score, F-1 score, recall, and lastly time comparison.

2 IMPORTANCE OF BIG DATA ANALYTICS FOR PREDICTIVE CLASSIFICATION

Importance of big data analytics is getting higher every day since the algorithms become more powerful to predict, classify and cluster any given data set. Importance of our case is any company can be used to predict individuals income to refer them goods in their income range or governments can provide additional support for the areas that have lower income range. There can be many possible things that can do with this kind of classification predictions.

3 DATA PREPARATION

We first used the pandas [28] to help to load the data in data frame format. This gave us a unique advantage, and faster processing of comma separated values for putting into data frame [48]. Our data consist of 15 variables. Some of these variables are continuous, and some of them are categorical variables, and our target variable was "income" attribute. After putting the data into data frames, we first got a statistical snapshot of continuous variables (age, education, capital gain, capital loss, hours worked) by using the pandas [27] functions as shown in Table 1.

	age	education	cap gain	cap loss	hours
count	32561	32561	32561	32561	32561
mean	38.581	10.08	1077.64	87.303	40.437
std.	13.640	2.572	7385.292	402.960	12.347
min.	17.0	1.0	0	0	1.0
25%	28.0	9.0	0	0	40.0
50%	37.0	10.0	0	0	40.0
75%	48.0	12.0	0	0	45.0
max	90.0	16.0	0	4356.0	99.0

Table 1: Statistical Summary of The Continuous Variables

3.1 Data Cleaning

After getting a snapshot from income data frame, we recognized that there is a column which has no meaning. The first task was to remove this entire column from our dataset we used pandas drop function for doing this task. After removing this column, we had more concise dataframe to analyze.

Moreover, removing the column we have encountered some missing values which labeled as "question marks" in data frame. In order to remove this values we first changed all the "question mark" values to "NaN" values by using pandas "replace" function [26]. After replacing all the question marks with "NaN" values, we used pandas missing value dropping function to remove all the "NaN" values from our dataset.

Furthermore, we start investigating the types of the variables, and in our case, we found two types of variable one of them labeled as "int64" which stands for integer values, other one labeled as object type of variable. From our previous example especially in "scikit-learn" it is better to use float object rather than "int64" for training the machine learning algorithms. Because their numerical output most of the time is "float64" object. We transferred all the

"int64" objects to "float64" objects. This was the last step of the cleaning process.

Our last process is changing the string values to numerical values on our target data which consist of string values ("$ 50K") for machine learning algorithms to understand this target data we need to transfer it to numerical values. Since we have only two categories, we will assign 1 and 0 as numerical values as shown in Table 2.

Description	Assigned Value
Individuals who makes more than $50K	1
Individuals who makes at or less than $50K	0

Table 2: Description of the Binary Values

Our shape of the data will also receive impact from changing to numerical. Our number of futures will go from 14 to 103. This is because we implemented one-hot-encode to our dataset. It is called one hot encoded because we transform the categorical variables into a more acceptable shape for the machine learning algorithms to perform well [45]. In other words "we implement binarization of the category to include as a future to train model [45]". As we can see in Table 3 and Table 4.

Company Name	Categorical Variable	Price
VW	1	2000
Acura	2	10011
Honda	3	50000
Honda	3	10000

Table 3: Example of One Hot Encoding Before [45].

VW	Acura	Honda	Price
1	0	0	20,000
0	1	0	10,011
0	0	1	50,000
0	0	1	10,000

Table 4: Example of One Hot Encoding After [45].

4 DATA EXPLORATION

After cleaning the data, we started our data exploration to learn little bit more from our data and make necessary changes if needed before putting into our machine learning algorithms. The first step in this process is getting the total count of the individuals as well as the count of the individuals who are making more than $50K and less than $50K which can be seen in below Table 5.

Moreover, we also look at the statistical values of each of the continuous variable we have. Those values given in Table 6. As we can see we have individuals who're age ranging from 17 to 90 years old with a mean of 38.58. If we look at the capital gains and capital losses, we have a standard deviation of 7385 and 402 respectively this is also another indication of skew in these variables.

Description	Count
Total Number of Individuals	30162
Individuals who makes more than $50K	7508
Individuals who makes at or less than $50K	22654

Table 5: Count of Income Variable Regarding to Individuals

	Age	Gain	Loss	Hours
Number of Instances	32,561	32,561	32,561	32,561
Mean	38.58	1077.64	87.303	40.437
Standard Deviation	13.640	7385.292	402.960	12.347
Minimum Value	17	0	0	1
25th percentile	28	0	0	40
50th percentile	37	0	0	40
75th percentile	48	0	0	45
Maximum Values	90	99999	4356	99

Table 6: Statistical Summary of Continuous Variables [44].

We used scatter matrix plot and applied the correlation function to see if we have any reliable correlation between any of the variables. As we can see from the correlation matrix Table 7 and correlation numbers Figure 1we do not have the high correlation between any variables. Correlation values range between -1 to 1. The correlation value of 1 is an indication of perfect positive correlation and correlation number -1 indicates a negative correlation between variables [15]. Because of lower correlation values, it will be tough to determine the classification by just looking at the correlations; this indicates we have sophisticated algorithms to determine the relationship between variables to classify individuals incomes.

	Age	Education	Capital Gain	Capital Loss	Hours Per Week
Age	1.0	0.043	0.080	0.060	0.101
Education	0.043	1.0	0.124	0.079	0.152
Capital Gain	0.080	0.124	1.0	-0.032	0.080
Capital Loss	0.060	0.796	-0.032	1.0	0.052
Hours Per Week	0.101	0.152	0.080	0.052	1.0

Table 7: Correlation Matrix [44].

Furthermore, we also explore the capital gains, capital losses, and hours per week variables which we used a histogram to plot the data into distribution form so we can see how all these attributes distributed. The reason we do the histogram is we want to see any skewness in our data. As shown in the histogram graphs in Figure 2 and Figure 3 in capital gains and capital loss we have highly skewed data which can cause issues later on in our algorithms. We apply a logarithmic function to do highly skewed data to less skewed [24]. Using logarithmic functions adds more value to data from the interpretable standpoint and "it helps to meet the assumptions of inferential statistics [24]".

Moreover, applying logarithmic function had an impact on distribution. We can see the changes on skew data in Figure 4 after applying logarithmic function.

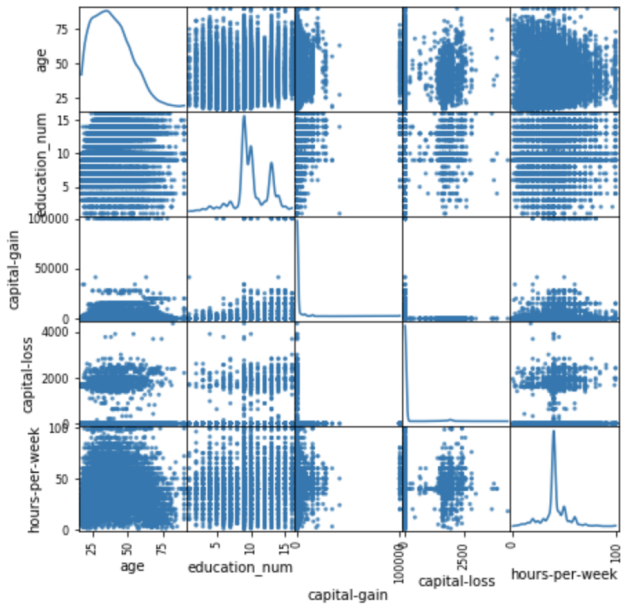

Figure 1: Scatter Matrix Plot [44].

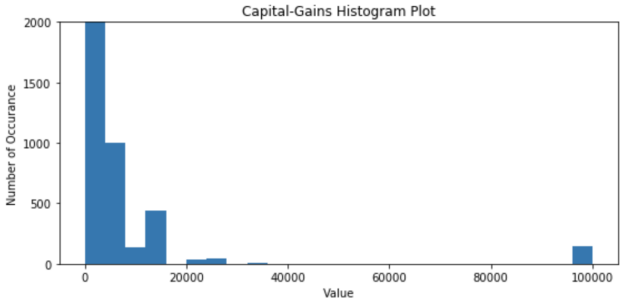

Figure 2: Histogram of Capital Gain [44].

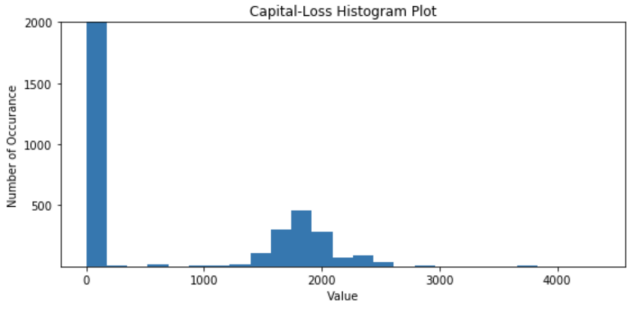

Figure 3: Histogram of Capital Loss [44].

5 MACHINE LEARNING ALGORITHMS TO CONSIDER

We have multiple algorithms to consider when we are doing the supervised learning. Each algorithm has its benefits and drawbacks.

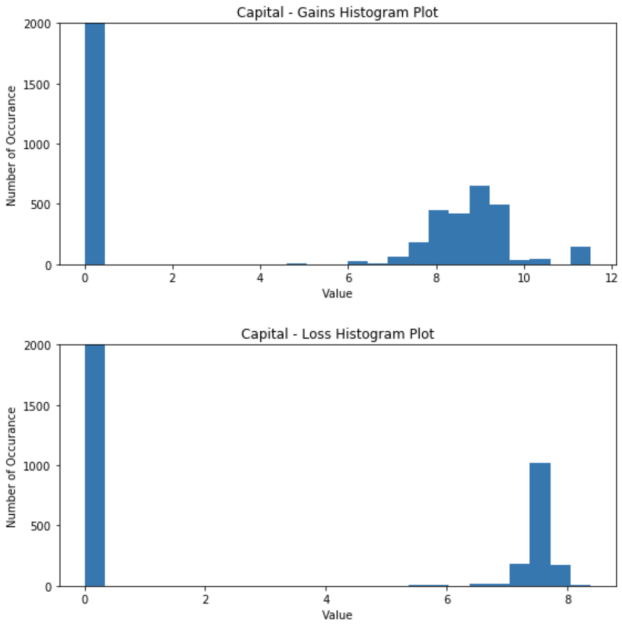

Figure 4: After Logarithmic Function Applied Histogram of Capital Gain [44].

We will consider several supervised machine learning algorithms for our predictions. The application we will use to implement these algorithms will be Python Scikit-Learn library. We will briefly explain each parameter included in these algorithms in Scikit-Learn.

First we'll look at the Scikit-Learn in Python framework we will go through the advantages in Scikit-Learn how we can implement any machine learning in just couple of simple line of codes in Scikit-Learn.

5.1 Why Scikit-Learn?

Scikit-learn developed by David Cournapeau in 2007. The development came from while he was working on summer code project for Google. After recognized and published by INRIA in 2010 project start the get more attention among worldwide. There are more than 30 active contributors and has secured several sponsorships from big technology companies[17]. "It also has a goal of providing common algorithms to Python users through consistent interface[2]". Scikit-Learn consists of several elements to make analytical predictions. These elements are shown below[23]:

Supervised Learning Algorithms: One of the most fundamental reason that Scikit-Learn's popularity comes from highly available supervised learning algorithms. These algorithms vary from regression models to decision trees and many more[23].

Cross Validation: Scikit-Learn includes various techniques to check the accuracy or any statistical measure between training and unseen testing set[23].

Unsupervised Learning Algorithms: Scikit-Learn had also various algorithms to support many unsupervised algorithms some of these include clustering, factor analysis, and neural network analysis[23].

Various example data-sets: Scikit-Learn comes with different data sets included in its package so users can start learning Scikit-Learn without the need of any data-sets[23].

Feature extraction: It has rich feature for extracting images or text from data-sets[23].

Algorithms that we will investigate shown below; we will go more deep analysis on each of these algorithms.

- Gaussian Naive Bayes
- Logistic Regression
- K-Nearest Neighbors (KNN)
- Stochastic Gradient Descent Classifier
- Support Vector Machines
- Decision Trees

5.2 Gaussian Naive Bayes

Naive Bayes bring many beneficial features; it is widely popular among machine learning applications[41]. The popularity of Naive Bayes comes from being able to handle large projects and data-sets faster than most algorithms[41]. It also can handle complex data-sets with categorical and non-categorical inputs [41]. Naive Bayes based on probabilistic classifier of Bayesian theory. It is also a favorite way of doing text categorization [46].

Term naive comes from it is the method of use probability among categories which assumes of independence among given class of attributes as shown in Figure 5. In other words, if we try to classify individuals from their email communications it will not take the order of words into account. Whereas in the English language we can tell the difference between sentence makes sense or not if we randomly re-order our words in the sentences. So it does not understand the text, it only looks at word frequencies as a way to do the classification. This is why it is called "Naive".

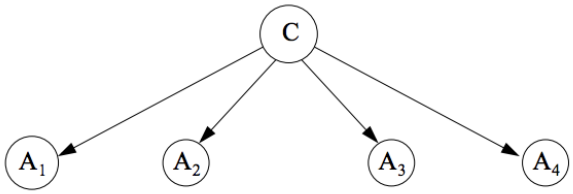

Figure 5: Example of Naive Bayes [50].

As we state above Naive Bayes derives from Bayesian Theory where the dimensionality of inputs is relatively high. Bayesian Theorem is stated below [16].

$$P(C \mid X) = \frac{P(X \mid C) \times P(C)}{P(X)} \tag{1}$$

Naive Bayes Classifier works as follows [16]:

Advantages of Naive Bayes [16]:

- Faster classification time for training data-set.
- Because of independent classification it improves classification performance.
- Performance is relatively good.

Disadvantages of Naive Bayes[16]:

- Often it requires a large number of data-sets to give adequate results.
- On some occasions which are relative to data-sets, it can give less accuracy.

5.3 Logistic Regression

Logistic Regression widely used for predicting "probability of failure in a given system, product, and process [34]". Logistic Regression also used in natural language analysis, it is an extension of conditional random fields [34]. It works as a classifier which learns the features from the input given and classifies them by multiplying the input value with the weight value [14].

$$P(C \mid X) = \sum_{i=1}^{N} W_i \times f_i \tag{2}$$

Main reason that Logistic Regression differs from Linear Regression is output variable for Logistic Regression is binary whereas output variable in Linear Regression is discrete(continuous) [12].

Advantages of Logistic Regression:

- It does not have any assumptions over distribution of classes [18].
- It is fast to train [18].
- Logistic Regression has fast classifying method of unknown data [18].
- We can easily extend to other regression for multiple classes like multinomial regression [18].

Disadvantages of Logistic Regression:

- One of the disadvantages of linear regression is it is not providing flexibility in some instances. What we mean by the " lack of flexibility is the linear dependency, and linear decision boundary in the instance space is not valid [42]". This disadvantage can be improved changing from Logistic Regression to Choquistic Regression[42].
- Logistic regression can provide poor results when there are more complex relationships in data [9].
- Logistic models also have over-fitting problems which come from a result of sampling bias [31].
- Because of Logistic Regression's predictions comes from the independent variable if the researcher includes wrong independent variables then model's prediction will have no value [31].
- Because it is predictions based on 1 and 0 model will have poor performance when predicting continuous variables [31].

5.4 K-Nearest Neighbors (KNN)

K Nearest neighbor has been primarily studied, and this popularity comes from it has been applied to many applications some of these applications are "spatial databases, pattern recognition, geographic information, image retrieval, computer game, and many other applications [29]". Due to an increase of mobile devices and people tends to use of applications like navigation K-nearest neighbor found itself another widely used area of location-based services due to an ability to found a target location [29].

Intuition behind the K Nearest Neighbor can be described as follows: " for a set P of n objects and a querying point q, return the k objects in P that are closest to q [29]."

Advantages of K Nearest Neighbors:

- K Nearest Neighbor is a basic and simple approach to implement [35].
- K Nearest Neighbor can perform well and efficiently with the large amount of data [43].
- K nearest Neighbor also does effectively well with noisy data sets ("if the inverse square of weighted distance used as the distance [43]"). In other words, it is flexible to feature and distance choices [35].

Disadvantages of K Nearest Neighbors:

- K Nearest Neighbor typically require large dataset to perform well [35].
- Time complexity could be high due to computing distance of each query to all training data points [43]. This time might be improved with some indexing (K-D Tree) [43].
- Determining the value of K can be time-consuming [43].
- It can be unclear to know which type of distance to use, as well as which variability to use to get the optimal results [43].
- Switching the different K values can result in the predicted class labels [30].

Many of these disadvantages are improving with the help of parallel distributed computing. Recent improvements in MapReduce framework allows users to run KNN algorithms in the cluster which had a significant effect on reducing the computation time [19].

Another area of improvements on KNN, is to implement different mapping functions such as kernel KNN, kernel difference weighted KNN, adaptive quasi-conformal kernel nearest neighbor, angular similarity, local linear discriminant analysis, and Dempster-Shafer [10].

5.5 Decision Trees

Decision Tree is another widely used algorithm model for classification and regression. Decision Trees uses a recursive split model where each recursive split is identified by each data point; this is an example of non-parametric hierarchical model [13].

Representation of decision trees is as follows; we sort the instances from root to leaf nodes, this sorting gives insights about the classification of the instance, every outcome descending from the root node corresponds to possible values for that variable [33]. We can classify an instance by starting from the root node and checking the attributes labeled on that node and moving down from that node based on attribute given attribute values [33] as shown in Figure 6.

Advantages of Decision Trees:

- Decision Tree applications are easy to interpret and understand [32]. This ease comes from their schematic representation [32]. Interpretation between alternatives can be expressed with single numerical number which is the expected value (EV) [32].

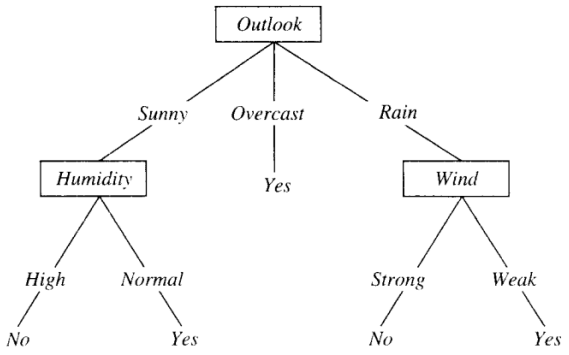

Figure 6: Example of Decision Tree Construction[33].

- Decision Trees can handle noisy or incomplete data-sets [32]. In other words it requires little effort of data preparation because of it is flexibility [7].
- It can handle both nominal and numerical variables [32].
- It can be modified easily whenever the new information is available [32].
-

Disadvantages of Decision Trees:

- Because of it is a use of divide and conquer method they can demonstrate good performance if there are few attributes exists when the attributes level goes into large number decision tree become more complex which will result in poor performance [32].
- Decision Trees are also susceptible to training set which can give a result of over-fitting [32]. In other words, it can believe the training set completely which will give an abysmal performance on testing set.
- ID3 and C4.5 decision tree algorithms require discrete values as input data.

5.6 Stochastic Gradient Descent Classifier (SGD)

Stochastic Gradient Descent recently got became more popular because of it is large-scale learning ability in machine learning problems [11]. It is a useful and straightforward way approach of linear classifiers under convex problems which is Support Vector Machines or Conditional Random Fields [3]. The originality of SGD derives from "Stochastic Approximation" which is a work from Robinson and Manroe [5].

Advantages of Stochastic Gradient Descent:

- One of the advantage of stochastic gradient descent is, it is easy to implement [38].
- Stochastic Gradient Descent is also efficient because of each step only relies on a single derivative which makes the computational cost 1 / n than normal gradient descent [37].

Disadvantages of Stochastic Gradient Descent:

- Stochastic Gradient Descent can be required to have many iterations, and it also requires some hyper-parameters [38].

- Feature scaling is a practice which used in the standardization of range of independent variables [47]. SGD also used this feature scaling technique and it can be sensitive to feature scaling [38].
- Another drawback of Stochastic Gradient Descent is while using GPU they are hard to parallelize or distributing them using computer clusters [25].

5.7 Support Vector Machines

Support Vector Machines is fallen under the classification methods in machine learning [6]. It is also a robust classification method that has been widely found itself an area ranging from pattern recognition to text analysis [6].

Fitting a boundary between data points is the principle of the support vector machines. This boundary divides the data points between classes, and each similar data point puts under the same class classification [6]. After training the support vector machines with training data-set, we only need to check whether the test data lies under the boundaries for testing set. Another thing to consider is after it creates the boundaries of the data remaining training data becomes obsolete because we only need the core set of points which supports the boundaries to classify the new data set. This core data points called "support vectors". It is called vector because of each data point contains a row of observed data values for attributes [6].

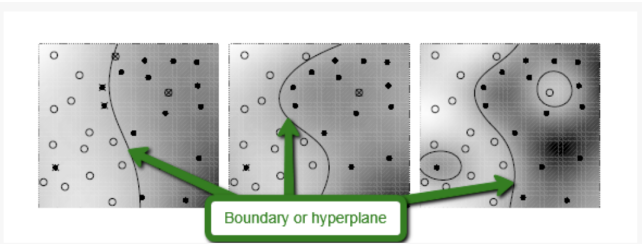

Figure 7: Example of Shows the Hyperplanes [6].

Traditionally boundaries are called "hyperplanes" and it is used to describe boundaries in more than three dimensions because they are hard or sometimes impossible to visualize.[7]. Figure 7. Optimality of hyperplane expressed as a linear function which requires maximum distance between the identified classes. It only considers a small number of training example to build this hyperplane. SVM hyperplanes based on " separation of positive (+1) and negative (-1) with the largest margin [39]".

One of the main characteristic of the machine learning is to generalization. In other words, we want to give a general idea that tends to fit any of our testing datasets optimally. Support vector machines are a perfect regarding generalizations because once the training data fitted by the support vector machines other than support vector data inside the training data becomes redundant which means that even with the small changes inside the data will not have a significant effect on general boundaries [6].

Advantages of Support Vector Machines:

- Generalizes the data well with the help of boundaries. Which reduces the overfitting [6].
- Classification accuracy in basic support vector machine will yield a 95 percent accuracy with a default settings [6].
- SVM can deliver a unique solution, because of optimality solution is convex. This will give an advantage over Neural Networks which has multiple solutions in local minima [1].

Disadvantages of Support Vector Machines:

- One common disadvantage of SVM, is the lack of transparency because of its non-parametric techniques [1].
- Another biggest disadvantage of SVM is it requires high algorithmic complexity and high level of memory for the large-scale implementations [39].
- According to Burgees, biggest limitation of the SVM is in the choice of kernel [4].

5.8 Ensemble Methods

Ensemble methods goes into classification algorithm category, they are learning algorithms which uses weighted vote for it is prediction methods, in other words, it is learning rules over a small subset of data then we combine these rules which we learn from the small subset of data to make predictions and/or classification on the testing data [8]. The originality of the Ensemble method comes from Bayesian averaging, but with the recent algorithms include "Bagging, error-correcting, and boosting [8]".

Bagging refers to simply the looking at data-sets and dividing the data-set to it is small subsets then learning the rules of that particular small subset. Next step is combining each learned rule from subsets to apply to more significant data set. Combining method mostly done with averaging the learned rules. Bagging also does better on testing set than standard Linear Regression analysis and linear regression does better on training set especially in third order polynomial [8].

Stacking

Boosting is another method used in Ensemble Methods. The difference from bagging is in boosting we need to pick subsets or examples that we are not good at in other words hardest examples. Then we combine these learned rules with the weighted mean instead mean used in bagging method.

Boosting is little different then bagging.

Advantages of Ensemble Methods:

- Prediction of the ensemble methods is better than most of the algorithms because of the combining methods intuition makes the model less noisy [36].
- They are more stable than other algorithms. [36]

Disadvantages of Ensemble Methods:

- Over-fitting may cause some disadvantages for ensemble learning but bagging operation will reduce this overfitting [36].

6 FITTING DATA INTO MACHINE LEARNING ALGORITHMS

In this section, we will show the techniques we used on the execution of the prepared data into machine learning algorithms. Before fitting the data into the machine learning algorithms, we split the data into two sets. These sets are the training set and the testing

set. We do splitting because of gaining an access of the future data will most likely be hard before future occurs, and because of this fact, it is a good idea to test our model with a dataset which our model has not seen it [40].

We used scikit-learn for splitting data into train and test we saved 20% of data for testing purposes as shown in Table 8 .

Splitting the Data	Sample Size
Training	24129
Testing	6033

Table 8: Train-Test-Split [44].

Furthermore, after splitting the data we put all of our training data into to each of the machine learning algorithm to get their prediction results. We also provided code at the beginning and the end of each algorithm to calculate their running time.

Before we move further we need to discuss critical characteristics of a machine learning algorithm. These are;

- Confusion Matrix
- Accuracy
- Recall
- F-1 Score
- Precision

6.0.1 ***Confusion Matrix:*** Confusion matrix develops from 4 key elements. These elements are true positive, true negative, false negative, and false positive. As shown in Figure 8 about the constructing a confusion matrix. If we want to build a confusion matrix by targeting individuals who are making more than $50K our true positive, true negative, false positive, and false negative explained below.

	Predicted class		
Actual Class		Class = Yes	Class = No
	Class = Yes	True Positive	False Negative
	Class = No	False Positive	True Negative

Figure 8: Example of Confusion Matrix Construction [20].

True Positive (TP): We can explain true positive as if the individuals make more than $50K and our model correctly classifies them as individuals who makes more than $50K, then this individual is in higher income range, in this case, we call it a true positive [20].

True Negative (TN): Intuition of true negative is if an individual makes less than $50K and our model correctly classifies them as individuals who makes less then $50K, then this individual is in lower income range. We call this true negative [20].

False Negative (FN): When an individual makes less than $50K and our model incorrectly classifies them in higher income range by making a mistake causes a false negative to happen [20].

False Positive (FP): When an individual is making more than $50K and our model classifies them in lower income range by mistake. This is called false positive [20].

6.0.2 ***Accuracy:*** Accuracy answers the question of how good is the model is. In our case this question will be out of all the individuals, how many did the models classify the individuals correctly. The mathematical expression of the accuracy is the ratio between the number of correctly classified points and the number of total points. We can think that if we have high accuracy, our model is excellent, but this is only where we have identical false positive and false negative values in our dataset [20].

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{3}$$

6.0.3 ***Precision.*** Precision answers the questions of out of all the points predicted to be positive how many of them were actually positive? If we translate this question into our case, we will have out of all the individuals that we are classified as lower income how many were actually have lower income. Higher precision indicates that we have low false positive rate [20]. Mathematical expression of precision is;

$$Precision = \frac{TP}{TP + FP} \tag{4}$$

6.0.4 ***Recall (Sensitivity).*** Recall answers the question of "out of the points that are labeled positive how many of them were correctly predicted is positive ? ". If we translate this to our case, we will have "out of the points that are labeled higher income how many of them correctly predicted is in higher income range ? ". Mathematical expression of the recall is;

$$Precision = \frac{TP}{TP + FN} \tag{5}$$

6.0.5 ***F-1 Score.*** The F-1 score is the idea of giving a decision by looking at only one score which will include precision, and recall scores. We cannot just take the average of precision and recall because if either of them is very low. We need a number to be low, even id the other one is not. This will leads us to look at the harmonic mean, and it works as follow. Let's say we have two numbers X and Y. X is smaller than Y, and we have the arithmetic mean, and it always lies between X and Y. It is a mathematical fact that the harmonic mean is always less than the arithmetic mean which is closer to the smaller number than to the higher number. Mathematical expression of F-1 score is;

$$F1Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{6}$$

6.1 Results

Now we can look at the results from each of the machine learning algorithm. Results also showed in Table 9 with the visualization of Figure 10. We can also see the running time of the each of the algorithm in Figure 9. Support Vector Machines is the winner for the highest running time for training the algorithm.

6.1.1 ***Naive Bayes.*** As shown in the Figure 10 we have a comparison of several supervised machine learning algorithms on our dataset. We can see that from the accuracy standpoint Naive Bayes algorithms have the lowest score which means that it did not do a good job for labeling true positives regards to all data but it did a good job in precision standpoint while doing a bad classification

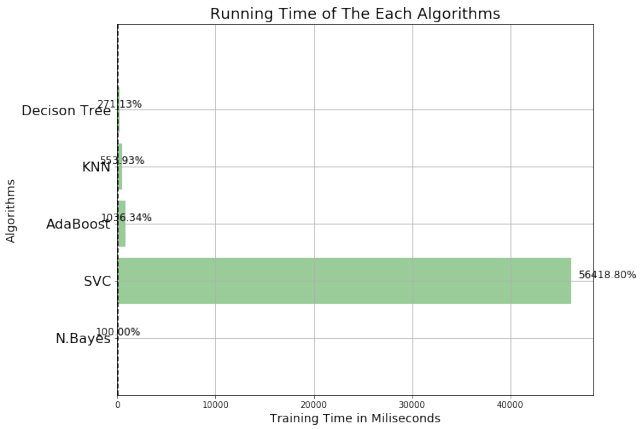

Figure 9: Supervised Learning Algorithm Running Time Results [44].

Name	Accuracy	Recall	Precision	F1 Score
Naive Bayes	0.4442	0.4642	0.9680	0.3053
SVC	0.8301	0.5969	0.5056	0.7284
AdaBoost	0.8499	0.6724	0.6189	0.7361
KNN	0.8184	0.6090	0.5682	0.6561
Decision Tree	0.8161	0.6231	0.6109	0.6459

Table 9: Results of the Algorithms [44].

from recall standpoint. Two key element for us in this situation is accuracy and f1 score(which consist of precision and recall).

6.1.2 ***Support Vector Machine***. Support Vector Machine is the second best algorithm in our case. This algorithm did very well job on classification it has the second highest accuracy and f1 score.

6.1.3 ***AdaBoost***. As we stated before ensemble algorithms learn from the small portion of the data and combine these learning to do the predictive task. As shown in Figure 10 adaboosting has the highest accuracy score among all the other algorithms. This algorithm should be our first choice to do predictive modeling. We believe that there is still an improvements on accuracy

6.1.4 ***K-Nearest Neighboors***. K-Nearest Neighbor algorithm in our project we set the k value to 5. K Nearest Neighbor algorithm also did a good job by placing itself third in accuracy score.

6.1.5 ***Decision Tree***. Decision Tree is gave a good accuracy but fall behind on f1 score as shown in Figure 10.

7 CONCLUSION

We presented the importance of analytical approach with machine learning algorithms and how they can be used to predict or classify the individuals with many different attributes like age, education, income, etc. We also presented weaknesses and strengths of these algorithms along with their precision, accuracy, recall, and F-1 scores by presenting with the visualizations. We also demonstrated the running time for each algorithm while using big data sets.

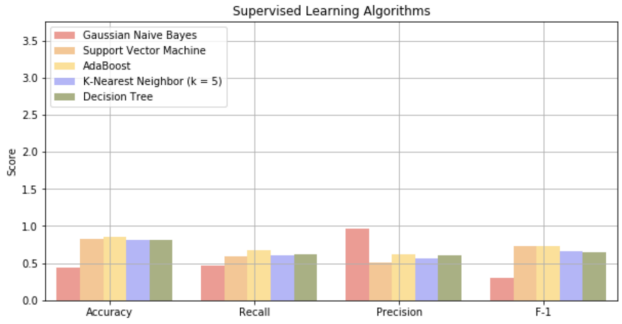

Figure 10: Supervised Learning Algorithm Results [44].

The source code of this project can found Github website which presented in reference section [44].

ACKNOWLEDGMENTS

The author would like to thank Dr. Gregor von Laszewski for his support and suggestions to write this paper.

REFERENCES

[1] L. Auria and A. R. Moro. 2008. Support Vector Machines (SVM) as a Technique for Solvency Analysis. Online. http://www.diw.de/english/products/publications/discussion_papers/27539.html
[2] L. Ben. 2015. Six Reasons why I recommend scikit-learn. Online. (Oct. 2015). https://www.oreilly.com/ideas/six-reasons-why-i-recommend-scikit-learn
[3] L. Bottou. 2010. Stochastic Gradient Descent. Online. (2010). http://leon.bottou.org/projects/sgd
[4] C. J. C. Burges. 1998. A Tutorial on Support Vector Machines for Pattern Recognition. *Data Mining and Knowledge Discovery* 2, 2 (01 Jun 1998), 121–167. https://doi.org/10.1023/A:1009715923555
[5] N. Deanna, S. Nathan, and W. Rachel. 2016. Stochastic gradient descent, weighted sampling, and the randomized Kaczmarz algorithm. *Mathematical Programming* 155, 1 (01 Jan 2016), 549–573. https://doi.org/10.1007/s10107-015-0864-7
[6] B. Deshpande. 2013. When do support vector machines trump other classification methods. Online. (Jan. 2013). http://www.simafore.com/blog/bid/112816/When-do-support-vector-machines-trump-other-classification-methods
[7] B. Desphpande. 2011. 4 key advanatages of using decision trees for predictive analytics. Online. (July 2011). http://www.simafore.com/blog/bid/62333/4-key-advantages-of-using-decision-trees-for-predictive-analytics
[8] G. T. Dietterich. n.d.. Ensemble Methods in Machine Learning. (n.d.). http://web.engr.oregonstate.edu/~tgd/publications/mcs-ensembles.pdf
[9] EliteDataScience. 2016. Modern Machine Learning Algorithms: Strengths and Weaknesses. Online. (May 2016). https://elitedatascience.com/machine-learning-algorithms
[10] O. F. Ertugrul and M. E. Tagluk. 2017. A novel version of k nearest neighbor: Dependent nearest neighbor. *Applied Soft Computing* 55, Supplement C (2017), 480 – 490. https://doi.org/10.1016/j.asoc.2017.02.020
[11] M. Fan. n.d.. How and Why to Use Stochastic Gradient Descent? (n.d.). http://anson.ucdavis.edu/~minjay/SGD.pdf
[12] J. Fang. 2013. Why Logistic Regression Analyses Are More Reliable Than Multiple Regression Analyses. *Journal of Business and Economics* 4, 7 (July 2013), 620–633. http://www.academicstar.us/UploadFile/Picture/2014-6/201461494819669.pdf
[13] M. A. Hassan, A. Khalil, S. Kaseb, and M. A. Kassem. 2017. Potential of four different machine-learning algorithms in modeling daily global solar radiation. *Renewable Energy* 111, Supplement C (2017), 52 – 62. https://doi.org/10.1016/j.renene.2017.03.083
[14] S. T. Indra, L. Wikarsa, and R. Turang. 2016. Using logistic regression method to classify tweets into the selected topics. *2016 International Conference on Advanced Computer Science and Information Systems (ICACSIS), Advanced Computer Science and Information Systems (ICACSIS), 2016 International Conference on* 1, 385-389 (2016), 385. http://proxyiub.uits.iu.edu/login?url=https://search-ebscohost-com.proxyiub.uits.iu.edu/login.aspx?direct=true&db=edseee&AN=edseee.7872727&site=eds-live&scope=site
[15] Investopedia. n.d.. Correlation Coefficient. Online. (n.d.). https://www.investopedia.com/terms/c/correlationcoefficient.asp
[16] D. S. Jadhav and H. P. Channe. 2014. Comparative Study of K-NN, Naive Bayes and Decision Tree Classification Techniques. *International Journal of Science and*

Research (IJSR) 5, 1 (Jan. 2014), 1842–1845. https://www.ijsr.net/archive/v5i1/NOV153131.pdf

[17] B. Jason. 2014. A gentle introduction to Scikit-Learn: Python Machine Learning Library. Online. (April 2014). https://machinelearningmastery.com/a-gentle-introduction-to-scikit-learn-a-python-machine-learning-library/

[18] H. Jeff. 2012. Introduction to Machine Learning. Online. (Jan. 2012). http://courses.washington.edu/css490/2012.Winter/lecture_slides/05b_logistic_regression.pdf

[19] J. Jiaqi and Y. Chung. 2017. Research on K nearest neighbor join for big data. In *2017 IEEE International Conference on Information and Automation (ICIA)*. IEEE, Department of Computer Engineering Wonkwang University Iksan 54538, Korean, 1077–1081. https://doi.org/10.1109/ICInfA.2017.8079062

[20] R. Joshi. 2016. Accuracy, Precision, Recall, and F1 Score: Interpretation of Performance Measures. Online. (Sept. 2016). http://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures

[21] R. Kohavi. 1996. Improving the Accuracy of Naive-Bayes Classifiers: A Decision-tree Hybrid. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD'96)*. AAAI Press, Silicon Graphics, Inc, 202–207. http://dl.acm.org/citation.cfm?id=3001460.3001502

[22] R. Kohavi and B. Becker. n.d.. Predicting whether income exceeds $50K/yr based on census data. Online. (n.d.). https://archive.ics.uci.edu/ml/datasets/Census+Income

[23] J. Kunal. 2015. Scikit-Learn in python - The most important Machine Learnig Tool I learnt last year. Online. (Jan. 2015). https://www.analyticsvidhya.com/blog/2015/01/scikit-learn-python-machine-learning-tool/

[24] M. D. Lane. n.d.. Log Transformations. Online. (n.d.). http://onlinestatbook.com/2/transformations/log.html

[25] V. Q. Le, J. Ngiam, A. Coates, A. Lahiri, B. Prochnow, and A. Y. Ng. 2011. On optimization methods for deep learning. In *International Conference of Machine Learning*. Stanford University, International Conferenfe of Machine Learning, Stanford University, NA. https://cs.stanford.edu/~acoates/papers/LeNgiCoaLahProNg11.pdf

[26] Pandas Library. n.d.. Dataframe replace. Online. (n.d.). https://pandas.pydata.org/pandas-docs/stable/generated/pandas.DataFrame.replace.html

[27] Pandas Library. n.d.. Pandas Dateframe describe. Online. (n.d.). https://pandas.pydata.org/pandas-docs/stable/generated/pandas.DataFrame.describe.html

[28] Pandas Py Data Library. n.d.. Pandas for Python. Online. (n.d.). https://pandas.pydata.org/

[29] L. J. Moon. 2017. Fast k-Nearest Neighbor Searching in Static Objects. *Wireless Personal Communications* 93, 1 (01 Mar 2017), 147–160. https://doi.org/10.1007/s11277-016-3524-1

[30] G. Nick. 2014. KNN. Online. (April 2014). http://www.nickgillian.com/wiki/pmwiki.php/GRT/KNN

[31] R. Nick. NA. The Disadvantages of Logistic Regression. Online. (NA). http://classroom.synonym.com/disadvantages-logistic-regression-8574447.html

[32] C. Petri. 2010. Decison Trees. Online. (2010). http://www.cs.ubbcluj.ro/~gabis/DocDiplome/DT/DecisionTrees.pdf

[33] U. Princeton. NA. Decision Tree Learning. Online. (NA). http://www.cs.princeton.edu/courses/archive/spr07/cos424/papers/mitchell-dectrees.pdf

[34] S. A. Raj, L. J. Fernando, and S. Raj. 2017. Predictive Analytics On Political Data. Congress. *World Congress on Computing and Communication Technologies* 10, 1109 (2017), 93–96.

[35] M. Ray. 2012. Nearest Neighbours: Pros and Cons. Online. (April 2012). http://www2.cs.man.ac.uk/~raym8/comp37212/main/node264.html

[36] S. Ray. 2015. 5 Easy Questions on Ensemble Modeling Everyone Should Know. Online. (Jan. 2015). https://www.analyticsvidhya.com/blog/2015/09/questions-ensemble-modeling/

[37] J. Rie and Z. Tong. 2013. Accelerating Stochastic Gradient Descent using Predictive Variance Reduction. In *Advances in Neural Information Processing Systems 26*, C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger (Eds.). Curran Associates, Inc., Rutgers University, New Jersey, USA, 315–323. http://papers.nips.cc/paper/4937-accelerating-stochastic-gradient-descent-using-predictive-variance-reduction.pdf

[38] Scikitlearn. n.d.. Stochastic Gradient Descent. Online. (n.d.).

[39] K. N. Shrivastava, P. Saurabh, and B. Verma. 2011. An Efficient Approach Parallel Support Vector Machine for Classification of Diabetes Dataset. *International Journal of Computer Applications in Technology* 36, 6 (Dec. 2011), 19–24. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.259.3757&rep=rep1&type=pdf

[40] D. Steinberg. 2014. Why Data Scientist Split Data into Train and Test. Online. (March 2014). https://info.salford-systems.com/blog/bid/337783/Why-Data-Scientists-Split-Data-into-Train-and-Test

[41] K. B. Tapan. 2015. Naive Bayes vs Logistic Regression: Theory, Implementation and Experimental Validation. *Inteligencia Artificial, Vol 18, Iss 56, Pp 14-30 (2015)* 1, 56 (2015), 14. http://proxyiub.uits.iu.edu/login?url=https://search-ebscohost-com.proxyiub.uits.iu.edu/login.aspx?direct=true&db=edsdoj&AN=edsdoj.0e372b34c5d48bcb72cd437eede1fd1&site=eds-live&scope=site

[42] A. F. Tehrani, W. Cheng, and E. Hullermeier. 2011. Choquistic Regression: Generalizing Logistic Regression Using the Choquet Integral. Online. (July 2011). https://www-old.cs.uni-paderborn.de/fileadmin/Informatik/eim-i-is/PDFs/Talk.EUSFLAT.11.pdf

[43] K. Teknomo. 2017. K-Nearest Neighbor Tutorial. Online. (2017). http://people.revoledu.com/kardi/tutorial/KNN/Strength%20and%20Weakness.htm

[44] E. B. Usifo. 2017. Income Prediction. Github. (Dec. 2017). https://github.com/bigdata-i523/hid343/tree/master/project

[45] R. Vasudev. n.d.. What is One Hot Encoding? do you have to use it ? Online. (Aug. n.d.). https://hackernoon.com/what-is-one-hot-encoding-why-and-when-do-you-have-to-use-it-e3c6186d008f

[46] Wikipedia. 2017. Naive Bayes. Online. (Nov. 2017). https://en.wikipedia.org/wiki/Naive_Bayes_classifier

[47] Wikipedia. NA. Feauture Scaling. Online. (NA). https://en.wikipedia.org/wiki/Feature_scaling

[48] Wikipedia. n.d.. Comma Seperated Values. Online. (n.d.). https://en.wikipedia.org/wiki/Comma-separated_values

[49] Wikipedia. n.d.. Decision Trees. Online. (n.d.). https://en.wikipedia.org/wiki/Decision_tree

[50] H. Zhang. 2004. *The Optimality of Naive Bayes.* resreport. University of New Brunswick. http://www.cs.unb.ca/~hzhang/publications/FLAIRS04ZhangH.pdf