Predicting An Individual's Income

Connor Maas

December 10th, 2021

ABSTRACT

An individual's lifestyle is almost always defined by the amount of money they make. This makes knowledge about one's income extremely powerful, as it gives insight to other valuable information about their life, such as their activities, lifestyle, and behavior. This paper uses machine learning classification methods to predict if a given individual's income is above or below 50,000 USD based on demographic data originating from the United States Census Bureau. I have constructed a new model, which has surpassed (in accuracy) previously researched models on the same dataset.

1.1 | INTRODUCTION

An individual's income is not just a sensitive topic that is impolite to bring up at the dinner table; income is a valuable piece of information about a person, which can be used for many different purposes. First and foremost, income often determines a person's lifestyle. Unless someone has inherited a large amount of money or was lucky enough to win the lottery, their income determines where they live, how they commute, what activities they participate in, and even specifics such as what they eat or what cleaning products they use. In almost all cases, people with high incomes are better off in every category. "Greater income is associated with increased happiness and well-being because of the greater advantages you have" [1], according to the well-being blog, MentalHelp. Since the knowledge of income is so powerful, the question becomes: is it good or bad information to have?

The answer is both. Organizations such as governments or charities can use income predictions to offer help to certain struggling communities. For example, the United States Census Bureau claims that they "calculate poverty status and ask about age and disability status" for helping "communities ensure older people receive appropriate assistance, such as financial assistance with utilities" [2]. Using income information in this manner is a "good" act. However, large companies are developing "new ad targeting features [which] allow marketers to direct ads to people within specific income brackets" [3]. This is perhaps not so "good," as it could promote division between different income brackets, or defamiliarize people to different ways of living.

Holding this knowledge is extremely important and grants an individual or organization with this knowledge large amounts of power. My main purpose in writing this paper is to expose "good" people to this information so the power of income knowledge is used in the right ways. This paper provides a key to that knowledge, as it uses machine learning classification methods to predict whether a given individual's income is above or below 50,000 USD based on a series of factors (discussed in Section 2).

1.2 | RELATED WORKS

I am not the first person to study this topic (predicting income). In fact, I have come across three different sources that have used the exact same dataset in order to predict if a person's income is above or below 50,000 USD.

The first source, which achieved an "overall accuracy of 86%" [4] in predicting income, focused mainly on data exploration. The author Abhinav Singh offers valuable insight on how he achieved a high prediction accuracy. He explains in his study that "the data needs to be reshaped in order to aid exploration of the data" [4]. Singh engages in data analysis and manipulation for the majority of the study, as he believes that "a very crucial step before modeling is the exploration of the independent variables" [4]. Clearly, Singh is implying that training a model over sloppy data will not yield an accurate result. It is important not to get over-eagerly to receive your results because data requires manipulation before it is trained on.

The second source using the same dataset, which achieved a "highest accuracy of 0.8620118" [5], focused on algorithm analysis: finding and combining the best classifiers. The author, Paul Ra, began his paper by claiming that he has "tested a variety of different models: logistic regression, linear discriminant analysis, K-nearest neighbor, classification tree, random forest, and gradient boosting" [5]. Rather than looking deeply into the data from the beginning, Ra decided to perform an algorithm analysis by continuously adjusting his model based on the result of each experiment. An example of this is when he discovered that "despite issues such as gender wage gap, the models generated showed that gender and race were not very significant" [5]. This is a disconnection that the human mind may have missed when parsing the data on its own because we consider race and gender to be large parts of our identities. Singh and Ra had opposite approaches to building a model, but both were remarkably successful.

The final source using this same dataset achieved a highest performance accuracy of 86% [6]. The author, Lichman, focused on larger-scale structuring of data. The process that he used is described as follows: "First, we will split our dataset into two parts - training set and validation set. Then, we will train a logistic regression model on samples from the training set and finally evaluate its accuracy based on the validation dataset" [6]. Structuring data in this way is very effective because it allows for efficiency and clarity when executing experiments. Also worth noting, Lichman talks about his model's potential to allow "high-income customers [to] be, for instance, exposed to premium products" [6], which is something I discussed in Section 1.1.

In conclusion, other sources have covered many methods in predicting income, spanning from a focus on ground-level data, to analyzing algorithms, to structuring data on a large scale, and so on. However, these models failed to break far beyond the 86% accuracy marker. My goal in this paper would be to build a model which surpasses 86% performance accuracy.

1.3 | PROCEDURE OUTLINE

(1) This paper aims to formulate a model which can accurately predict if the income of a given individual is either above or below 50,000 USD per year (class value). This section elaborates on the general processes which takes place over the course of this paper and the sections in which specific information is contained.

(2) To make a prediction on income, the model is trained over a dataset derived from the United States Census Bureau, as discussed in Section 2.1. The features contained within the dataset, as well as the methods of handling the data (such as splitting it into development, cross-validation, final testing sets), are discussed in Section 2.2. This section, along with Section 2.3A and 2.3B, cover data analysis using two different machine learning applications, Weka and Lightside.

(3) Section 3 builds off Section 2 and displays the baseline performances for the selected classifiers which will be analyzed in further detail.

(4) Error analyses and parameter tuning are discussed in Sections 4.1 and 4.2 respectively. These two sections are key in developing the models to predict income with the highest accuracy.

(5) Finally, the updated models are run on the final test set (see Section 5) and the results are discussed in the following section (Section 6).

2.1 | DATA COLLECTION

This section covers the origin and reliability of the dataset [7]. The dataset originates from real and publicly available information from the United States Census Bureau in 1994. However, this data was compiled by Barry Becker (A General Manager for U.S. Alliances and Partnerships [8]). According to the United States Census Bureau, the data "comes from a variety of sources" [9], including but not limited to anonymous surveys, governmental and commercial entities, and online databases. While the Census Bureau may not be completely accurate in all cases, such as estimating population count, this dataset involves little prediction and is simply recorded data. The University of California Berkeley wrote an article about the reliability of the United States Census, claiming it is "one of the most reliable sources of U.S. demographics statistics and data" [10]. This is especially true considering the limited number of sources that have access to massive amounts of data on demographics and income. In conclusion, this dataset is reliable for the purpose of analysis.

2.2 | DATA PREPARATION

This section elaborates further on the dataset and also explains the process of transforming the data to make it useful to two machine learning applications, Weka and Lightside, where the data will be analyzed.

First, I narrowed down the number of instances because my dataset was extremely large. The total number of instances I had was 32,562 instances. Machine learning algorithms take much longer to run on data sets of this size and are not much higher in accuracy than sets of far fewer instances (run faster). As a result, I randomly selected[1] 1,250 instances from these 32,562.

The initial features, their type, explanations, and examples can be seen in Table 1 (next page).

[1]Assigned a random number to each instance, sorted these instances, and selected the first 1,250 instances.

Table 1:

Feature	Type	Explanation	Examples
age	Numeric	The amount of years an individual has lived	43, 82
workclass	Nominal	A general term used to categorize occupations	Private, self-employed-not-inc
fnlwgt	Numeric	“Final weight”, the number of people which an individual is believed represent according to census data	54878, 7322, 588932
Education	Nominal	The highest level of education which an individual has reached	HS-grad, Bachelors
Education-num	Numeric	A numeric representation of the highest level of education which an individual has reached	6, 9, 13
Marital-status	Nominal	The status of an individual's marriage history	Divorced, Married-civ-spouse
Occupation	Nominal	A specific category describing an individual’s occupation	Sales, Prof-specialty,
Relationship	Nominal	The individual's relationship to another individual (very general)	Husband, Not-in-family, Own-child
Race	Nominal	A term referring to an individual's racial background	Black, White
Sex	Nominal	The biological gender of an individual	Male, Female
Capital-gain	Numeric	USD amount of profit from investments	0, 7688
Capital-loss	Numeric	USD amount of loss from investment	0, 1887
Hours-per-work	Numeric	The number of hours an individual works per week	45, 60, 65
Native-country	Nominal	An individual’s native country (place of origin)	United States, Iran, Mexico
Income*	Nominal	An individual’s yearly income	<=50k, >50k

* Class value

As discussed in Section 1.1, income is a valuable piece of information, which can be useful to predict. Not only that, but income is also a nominal feature with only two values, which makes the prediction process simple. As a result, I chose income as my class value.

As can be seen in Table 1, eight of thirteen features are nominal. A high number of nominal features makes it difficult to improve the accuracy of the model in Lightside and Weka. Therefore, I decided to convert all of these nominal features into binary. This can be executed in Weka using the "nominalTobinary" unsupervised filter. This process yielded 97 different features (all numeric). I then standardized the original numeric features (the remaining five of thirteen) to get a proper comparison of the average cell value when doing my error analysis (discussed in Section 4.1).

Some features, explanations, and examples can be seen in Table 2 below.

Table 2:

Feature	**Type**	**Explanation**	**Example**
'workclass = private'	Numeric	1 if workclass is private, else 0	0,1
'workclass = State-gov'	Numeric	1 if workclass is private, else 0	0,1
'education = masters'	Numeric	1 if workclass is private, else 0	0,1
age (standardized)	Numeric	Standardized values of age	-0.026619, 0.73729
fnlwgt (standardized)	Numeric	Standardized values of fnlwgt	0.943722, -0.13139

The next step in this process is randomly dividing the two datasets (original and binary) into three separate sets: the development set (20% of instances), the cross-validation set (70% of instances), and the final testing set (10% of instances). The development set is used for technical analysis as well as testing the model. The final testing set is excluded from any training or analysis until the final test to make sure that the trained model is completely unbiased to the data. The cross-validation set is used both to train the model and for parameter tuning (discussed in Section 4.2).

To set up a proper parameter tuning procedure (discussed in Section 4.2), the binary cross-validation set needs to be equally partitioned into ten different sets (no overlapping data). Five of these sets are training sets and the remaining five are testing sets. The division process is executed in Weka using the unsupervised filter, "removeFolds," which splits the data (by fold) into testing and training sets. A model is trained on each of the training sets and tested on each of their corresponding testing sets (this process is further discussed in Section 4.2).

The following datasets can be visualized in Figure 1 below (I will refer to these sets as such in the following sections).

Figure 1:

Complete dataset	*(Dataset 0)*
| Reduced dataset	*(Dataset 1)*
| Development set	*(Dataset 1B)*
| Final testing set	*(Dataset 1F)*
| Reduced dataset (edited)	*(Dataset 2)*
| Development set (edited)	*(Dataset 2D)*
| Cross validation set (edited)	*(Dataset 2C)*
| Training 1 - 5 sets (edited)	*(Datasets 2CTr 1 - 5)*
| Testing 1 - 5 sets (edited)	*(Datasets 2CTe 1 - 5)*
| Final testing set (edited)	*(Dataset 2F)*

* An indentated set “X” under a set “Y” indicates that “X” is a subset of “Y” *

2.3A | DATA EXPLORATION (Cluster Analysis)

This section outlines my process for conducting a cluster analysis on Dataset 1B (development set). I have included this process to come to some initial conclusions about the data, which will allow for justifiable selections of classifiers in Section 2.3B.

I experimented with several different numbers of clusters. I noticed that as the number of clusters increased, the less exclusive the clusters became. Since eight of my features are nominal, I conducted a “classes to clusters analysis” for each number of clusters. As expected, I found that increasing the number of clusters increases the number of incorrectly clustered instances.

The performance is shown in Table 3 (next page).

Table 3:

Number of Clusters	Incorrectly Classified Instances
1	26.4
2	37.6
3	50
4	60.4
5	64.8
6	72.8
7	73.6
8	74.8

According to table 3, one cluster is the most effective way of grouping my data. However, this is overlooking the fact 73.6% of the instances in this dataset are "<=50k" income. Therefore, the model can just classify everything as "<=50" income and seemingly perform with high accuracy. However, looking into the data is important to find trends and regularities. I decided to choose 2 clusters because this offers some insight into the data, and it still allows for a high number of correctly clustered instances.

Table 4 (next page) contains the three random instances from each cluster. Since I have 13 features, I decided to cut the table to only include the relevant features for my comparison with the centroid cluster instances.

Table 4:

Age	Workclass	Education	Marital Status	Occupation	Sex	Income	Cluster
24	Private	7th-8th	Never	Handlers-cleaners	M	<=50k	1
26	Private	Some-college	Never	Adm-clerical	M	<=50k	1
35	Private	HS-grad	Never	Exec-managerial	F	<=50k	1
43	Private	Bachelors	Married	Prof-specialty	M	>50k	2
57	Self-employed	Prof-school	Married	Prof-specialty	M	>50k	2
44	Self-employed	HS-grad	Married	Exec-managerial	M	<=50k	2

Most of these features fall in line with the centroid instances for the cluster they were assigned to. For all cluster 1 instances, the working class was “private” and the marital status was “never married.” For one or more cluster 1 instances, the education was “some college,” the occupation was “Adm-clerical,” and the sex was “male.” For all cluster 2 instances, the sex was “male” and the marital status was “married.” For one or more cluster 2 instances, the working class was “private,” the occupation was “prof-specialty,” and the education was “HS-grad.”

Cluster 1 and 2 are both fairly homogenous. I noticed that: (1) Cluster 2 instances had older individuals than cluster 1 instances (2) Cluster 2 instances had a higher level of education than cluster 1 instances. (3) Cluster 1 instances were unmarried, whereas cluster 2 instances were married. (4) Cluster 2 had 2 individuals in professional specialties as occupation and 2 individuals as self-employed for workclass, while cluster 1 had neither of those (5) Only cluster 1 instances contained a female.

In conclusion (based off these instances), I would group the data into two categories:

- “Young, less-educated, unmarried males and females” (cluster 1)
- “Older, more-educated, professional, married males” (cluster 2)

2.3B | DATA EXPLORATION (Classifier Selection)

This section outlines the process for selecting classifiers to improve, considering their baseline performances and other knowledge. The ultimate goal of this paper (as discussed in Section 1) is to formulate a model that performs with the highest possible accuracy. Therefore, it is important to choose a classifier, which is both accurate and possible to improve.

I began by taking into context the data, rather than just looking at each instance as a group of values. In doing so, I realized that there will always be outliers in terms of income; there will always be individuals (instances) of completely different demographics and backgrounds making the same amount of money. This is because individuals are not just a collection of numbers and values; each individual has their own free will, so despite all demographics and backgrounds, an individual can make any feasible amount of money. For example in Section 2.3A (Table 4), there was an "exec-managerial, married, male" with an income of fewer than 50,000 USD per year (which did not conform to the conclusion I made in that section).

As a result of this, I chose the Locally Weighted Learning classifier (LWL). Locally weighted learning is the perfect classifier to ignore the anomalies and focus on weighing "local" instances. However, I was skeptical of too much or too little weight being placed on certain categories. Since I was able to identify seven different impactful features in Section 2.3A, I aimed to encourage the model to capture the bigger picture rather than hyper-focusing on a few heavily weighted instances. I also didn't want the model to become confused by looking at every category completely equally. As a result, I selected the WeightedInstancesHandlerWrapper (WIHW), which assists with handling weighted instances.

I also predicted the JRIP[2] classifier to be successful because using rules to classify should be extremely effective in interpreting the many different binary categories in Dataset 2C (edited cross-validation set). For example, as could be seen in the 2.3A conclusion, a specific combination of categories, such as "older, more educated, professional, married males" could become a rule. However, this would be conducted on a much larger scale with many different categories.

Finally, I evaluated the performances of NaiveBayes and Logistic to get a better understanding of how my selected classifiers compared to other common machine learning classifiers. In other words, I wanted to make sure my predictions were accurate, so I used these classifiers as references.

The performance of these four classifiers on default settings using a ten-fold cross-validation over Dataset 2D (binary development) is contained within Table 5 (next page).

[2] A classifier which uses ordered rule lists and reduced-error pruning.

Table 5:

Classifier	Correctly Classified Instances
JRIP	82.9714%
NaiveBayes	82.8571%
Logistic	80.8%
LWL (WIHW)	75.0857%

As evident in Table 5, JRIP was successful, leading all four classifiers in correctly classified instances. LWL (WIHW), conversely, had proven to be significantly less effective than the popular models. However, I was confident enough in my practical observations of the data and the cluster analysis that I attempted to improve this model. I believe that the default settings did not allow LWL (WIHW) to perform its best.

In conclusion, I proceeded in an attempt to improve both the JRIP and LWL (WIHW) classifiers.

3 | BASELINE PERFORMANCE

This section displays and discusses my baseline performances for the JRIP and LWL (WIHW) classifiers. Both models were trained over Dataset 2C (binary cross-validation) and tested on Dataset 2D (binary development). The results can be seen in Table 6 below.

Table 6:

Classifier	Correctly Classified Instances	Kappa Statistic
JRIP	86%	0.5905
LWL (WIHW)	76%	0

I had two specific observations from these baseline results: (1) The Kappa statistic of LWL (WIHW) is 0, meaning that the model is predicting all instances to be one value (<=50k or >50k). This will be important to note in Sections 4.1 and 4.2, which aim to improve the model. (2) JRIP's performance has increased significantly. There is room for improvement in both of the models (especially LWL (WIHW)).

4.1 | ERROR ANALYSIS

This section performs a complete error analysis using Lightside aimed to improve both the JRIP and LWL (WIHW) models. All models are trained on Dataset 2C (binary cross validation) and tested over the Dataset 2D (binary development).

I first used Support Vector Machines (SVM) in Lightside for my error analysis to quickly identify the 4 different problematic features. The initial performance of the support vector machines can be seen in table 7 below.

Table 7:

Classifier	Correctly Classified Instances	Kappa Statistic
Support Vector Machines	0.808	0.4676

ERROR 1:

The first problematic feature I identified was “education = HS-grad,” which often confused the model to predict a “<=50k” income rather than a correct “>50k” income. This feature had a high horizontal absolute difference of 0.2011, an average cell value of 0.2979, and a feature weight of 0.9089.

It makes sense that “education = HS-grad” (this means that the individual never attended college) is likely to predict “<=50k” because a lower level of education would indicate a lower probability of an income of “>50k” (it is easier to secure higher paying jobs with a stronger education). However, I believe that there are individual outliers (that attended high school, but not college), who fit a certain demographic and can be singled out to improve the model's accuracy.

After looking into the data, those with “>50k” income were married in 10/12 cases. In the “<=50k” cases, only 26/57 individuals were married. I also noticed that those with a “>50k” income were male in 11/12 cases, whereas there was a much more even distribution of males and females in the “<=50k” instances. Finally, I noticed that those making “>50k” were all white (race). Based on the data, I believed that it was a good indicator that an individual had a “>50k” income if they were currently married, male, and white.

In an attempt to combat this error, I have decided to create a new binary feature specifically designed for this group called “specific HS-grad.” In pseudocode, this category can be interpreted as shown in Figure 2 (next page).

Figure 2:
IF “education = HS-grad”
AND “marital-status = Married-civ-spouse”
AND “sex = Male”
AND “race = White”
THEN: 1
ELSE: 0

After testing the model with this new feature present, the performance can be seen below in Table 8.

Table 8:

Classifier	Correctly Classified Instances	Kappa Statistic
Support Vector Machines	0.8286	0.5243

I then compared this model with the baseline model to get a sense of the difference in performances. Since $0.05 < p^3 < 0.1$, the model showed marginal improvement. This is an important discovery and solution for this error, and will be important for improving the models performance. I plan on including it in my model for future testing.

ERROR 2:

The next problematic feature I identified was “hours-per-week,” which often confused the model to predict a “>50k” income rather than a correct “<=50k” income. This feature had a high horizontal absolute difference of 1.0001, an average cell value of 0.7689, and a feature weight of 0.3749.

One might think that an individual who works more hours will make more money. However, this is not always the case, as the type of occupation often determines the amount of money that an individual makes. The model was confused by the small group of people who worked low hours but still had “>50k” income. My goal was to target this group of people and make sure they received a “bonus” for being in their profession.

After looking at the data, I noticed that 15/25 individuals who worked 40 hours per week and had a “>50k” income were either “occupation = Prof-specialty” or “occupation = Exec-manager.” On the other hand, those who had an income of “<=50k” were much less likely to hold either of these positions.

In order to combat this error, I decided to add 10 hours to the “hours-per-week” column if the individual was occupied in a professional specialty or an executive/manager position. I

[3] The p-value is a statistical value which helps to determine if two models’ performances are different enough to be considered “significant.”

created a new feature “New hours per week” (which will replace the previous “hours-per-week” feature). In pseudocode, this category can be interpreted as shown in Figure 3 below.

Figure 3:
IF “occupation = Prof-specialty”
OR “occupation = Exec-managerial”
THEN: Hours-per-week + 10
ELSE: Hours-per-week

I standardized the values for this feature, creating the final feature to use in my training and model evaluation “New hours per week (standardized).” After testing the model with this new feature present, the performance can be seen in Table 9 below.

Table 9:

Classifier	Correctly Classified Instances	Kappa Statistic
Support Vector Machines	0.7943	0.3965

The new model did not improve, as I had aimed for, and instead decreased in both accuracy and Kappa. The improvement from this model to the baseline model is an insignificant improvement as $p^4 = 0.28$. Overall, the error analysis and solution were ineffective. I decided not to use this new feature going forward because it does improve the accuracy of the model (if anything, it decreases accuracy).

ERROR 3:

The next problematic feature I identified was “marital status = married-civ-spouse,” which often confused the model to predict income of “>50k” rather than a correct income of “<=50k.” This feature had a low vertical absolute difference of 0.0018, a high average cell value of 0.8727, and a high feature weight of 0.8254.

This error is difficult to interpret because I previously assumed that being married in almost all cases is a good indicator of making money (“>50k” income). However, the low horizontal absolute difference begged to differ. I immediately continued by looking further into the data for an explanation.

I noticed that non-elderly (<60) individuals with either capital gains or capital losses were more likely to have a “>50k” income. These individuals confuse the model to predict “>50k” in most cases for “marital status = married-civ-spouse” because of their surprising success.

[4] Ibid.

In order to remove these "exceptional" individuals, I decided to add a new binary feature, "Exceptional marriage," which is true if an individual is under 60 years old and has capital gains or losses. In pseudocode, this category can be interpreted as shown in Figure 4 below.

Figure 4:

IF "Marital-status = Married-civ-spouse"
    AND "age < 60"
    (AND "Capital-gains > 0" OR "Capital-losses" > 0)
THEN: 1
ELSE: 0

After testing the model with this new feature present, the performance can be seen in Table 10 below.

Table 10:

Classifier	Correctly Classified Instances	Kappa Statistic
Support Vector Machines	0.8126	0.4646

I then compared this model with the baseline model to get a sense of the difference in performance. Since $p^5 > 0.05$ ($p = 0.696$), the model showed insignificant improvement. However, it is notable that the model's accuracy and Kappa both increased slightly. I decided to proceed with caution when using this feature to improve the model because I believe that the small improvement will be more beneficial than harmful and it may be more effective on a different dataset.

ERROR 4:

The final problematic feature I identified was "relationship = husband," which often confused the model to predict an income of ">50k" rather than a correct income of "<=50k." This feature had a low vertical absolute difference of 0.0575, a high average cell value of 0.8025, and a high feature weight of 0.1398.

Since being a husband requires that you are married and being married (as a male) requires that you are a husband, the "marital-status" (Error 3) and "relationship" features go hand-in-hand. Because nominal features were converted to binary, the values of each nominal feature could be separately evaluated as their own binary features. There was no logical reason to keep "relationship = husband" as a feature in the dataset because it was both problematic and connected to an error that I previously attempted to solve.

[5] Ibid.

As a result, I decided to remove "relationship = husband" from the dataset completely. I then tested the new data set with the fix from error 3 (because they are related). The performance can be seen in Table 11 below.

Table 11:

Classifier	Correctly Classified Instances	Kappa Statistic
Support Vector Machines	0.8137	0.467

I then compared this model with the baseline model to get a sense of the difference in performance. Since p[6] > 0.05 (p = 0.696), the model showed insignificant improvement. Similar to Error 3, I decided to proceed with caution when using this feature to improve the model because I believe that the small improvement will be more beneficial than harmful and it may be more effective on a different dataset.

EVALUATION OF ERROR ANALYSIS:

I decided to compare the error analyses I conducted using SVM with the other error analyses I conducted using JRIP and LWL (WIHW). Since LWL (WIHW) had a Kappa statistic of zero, I was expecting to see the model completely favor one value. My prediction was correct, as the confusion matrix in Lightside showed that every single instance had been classified as "<=50k." This leaves a great opportunity for tuning, which is discussed in Section 4.2. For JRIP, I immediately noticed that the model had the exact same problematic features ("marital status = married-civ-spouse", "hours-per-week", "education = HS-grad", "relationship = husband") as SVM. This is great evidence to suggest these features are problematic across all classifiers. This confirmed my conclusion for this section:

- Add feature "Specific-HS-grad"
- Add feature "Exception-marriage"
- Remove "relationship = Husband."

[6] Ibid.

4.2 | PARAMETER TUNING

I tuned both the JRIP and the LWL (WIHW) models using Weka. I was able to do this by loading datasets CTr 1 - 5 (see Section 2.2) into the Weka experimenter. I then added the classifiers (JRIP and LWL (WIHW)) with different parameter settings to compare their performances on each fold.

JRIP TUNING:

Since JRIP has two parameters to tune (minNo and optimizations), there are many combinations which are possible to succeed. After testing different combinations of these two parameters, I noticed that the performance of JRIP on the dataset only decreased as minNo increased. Therefore, I left minNo at it's default value of 2.0 and focused on tuning optimizations.

The baseline performance with JRIP on default settings was 83.84% correctly classified. I then performed a standard tuning process (displayed in Table 13) by altering the optimization settings and looking for the highest performing model.

Table 12:
Optims. = "Percent correct: train set performance for optimizations = 'x', minNo = 2.0"
Test. perform. = "Percent correct: test set performance"
*default

Fold	Optims. = 2*	Optims. = 10	Optims. = 50	Optims. = 100	Optims. = 500	Optimal setting	Test perfom.
1	83.14	83.71	83.71	84.00	84.57	500	85.1429
2	82.14	81.43	81.43	81.43	81.43	2	83.4286
3	81.29	83.00	83.57	83.29	83.14	50	85.1429
4	84.71	86.43	86.29	86.14	86.57	500	80
5	83.43	83.57	84.14	84.14	83.29	50	84.5714

Average = (85.1429 + 83.4286 + 85.1429 + 80 + 84.5714) / 5 = **83.65716 correctly classified**

Both "optimizations = 50" and "optimizations = 500" occurred twice as the most optimal setting. I decided to consider "optimizations = 50" as the overall optimal setting because it occurred the most frequently (tie) and also did not take very long to run (500 optimizations was very slow). Despite tuning not showing statistical improvement, as $p^7 > 0.05$, I decided to include

[7] Ibid.

the optimal setting of "optimizations = 50" in my final model because it showed slight improvement.

LWL (WIHW) TUNING:

Next, I tuned LWL (WIHW), which was a standard one parameter-tuning process (displayed in Table 13). The baseline performance of LWL (WIHW) on default settings was 75.36% correctly classified.

Table 13:

KNN = "Percent correct: train set performance for KNN = 'x'"
Test. perform. = "Percent correct: test set performance"
*default

Fold	KNN = 0*	KNN = 10	KNN = 100	KNN = 500	Optimal setting	Test perfom.
1	74.59	78.40	80.95	74.59	100	82.8571
2	75.48	78.15	81.57	75.48	100	80
3	74.97	77.64	82.33	74.97	100	82.2857
4	75.48	77.14	81.84	74.46	100	76
5	74.46	78.27	80.93	74.46	100	83.8571

Average = (82.8571 + 80 + 82.2857 + 76 + 83.8571) / 5 = **80.99998 % correctly classified**

Tuning was beneficial, as p[8] < 0.05. Since the improvement was statistically significant, I decided to include the optimal setting of "KNN = 100" in my final model.

[8] Ibid.

5 | FINAL RESULTS

First, I trained over Dataset 2C (edited cross validation) with JRIP and LWL (WIHW) on default settings. I then tested on Dataset 2F (edited final test set). The performance can be seen below in Table 14.

Table 14:

Classifier	Correctly Classified Instances	Kappa Statistic
JRIP	84%	0.4843
LWL (WIHW)	80.8	0

I edited the Dataset 2F by adding the features “Specific-HS-grad” and “Exception-marriage”, as well as removing “relationship = Husband” (discussed in section 4.1). I then trained both JRIP (optimizations = 50, minNo = 2.0) and LWL (WIHW) (KNN = 100) over Dataset 2C. I used the newly edited Dataset 2F as a test set to yield my final results, which can be seen below in Table 15.

Table 15:

Classifier	Correctly Classified Instances	Kappa Statistic
JRIP	88.8%	0.6501
LWL (WIHW)	86.4%	0.5068

For both JRIP and Locally Weighted Learning (WeightedInstancesHandlerWrapper), the differenes in performance were statistically significant (increased), as $p^9 < 0.05$ (in both bases).

[9] Ibid.

6 | DISCUSSION

In this paper, I describe the process of successfully building two different models, which are able to accurately predict if an individual's income is above or below 50,000 USD. Beginning with a quick introduction on the relevance of this topic, I dove immediately into data analysis and processing. I decided to evaluate and improve two different classifiers: JRIP and Locally Weighted Learning (WeightedInstancesHandlerWrapper). I then conducted baseline experiments, produced several error analyses, and finally used parameter tuning to optimize my models. In the end, both models showed statistically significant improvements. Notably, in Section 1.2, I set a goal to surpass 86% accuracy, as three different referenced papers (who used my same dataset) were unable to. My JRIP model, achieving 88.8% accuracy, made a nearly 2% accuracy improvement from these previous models. My LWL (WIHW), achieving 86.4% accuracy, made significant improvements, after having a performance of just 75.09% in Section 2.3B.

Also in Section 1.2 of this paper, I discuss the previous methods which have been used to build models on this dataset. These methods played an extremely important role in the development of my ideas and in improving my model. Being focused on ground-level data allowed me to come to many small and important conclusions, while datasets structuring focused me toward a larger-scale formal analysis. For example, in Section 2.3B, I decided to continue trying to improve the worst-performing classifier, LWL (WIHW), despite NaiveBayes, JRIP, and Logistic being significantly more accurate. The reason for this is that I paid very close attention to the data in Section 2.3A during cluster analysis. I recommend that future researchers follow a similar process when studying similar data.

An interesting observation I made was that parameter tuning seemed to work extremely well, while error analysis did not. The highest improvement I was able to achieve from error analysis was a statistically marginal improvement. In the tuning, however, I saw significant performance increases with multiple different parameter settings. In conclusion, tuning is likely the better option for data involving people, their demographics, and their backgrounds.

Specifically, the LWL (WIHW) model had a huge statistical increase from its performance in Section 2.3B, which I would say is almost exclusively due to the tuning. Although the JRIP model had a significant increase in performance, its improvement over the course of the paper was not nearly as notable as the LWL (WIHW) model. I would attribute this to JRIPs apparent effectiveness as an algorithm in general, which makes it hard to improve significantly.

A limitation to this paper is that the model doesn't have a literal application. That is, predicting a specific individual's income, such as your friend or business partner, is not a practical use for this model. Rather, the model is effective for forming groups of people and labeling each of those groups as an "almost all <=50k" or "almost all >50k." Something else to consider is that there are a limited amount of datasets which contain both an individual's income and their demographics. Therefore, it is hard to find new data leads at any point in time which will significantly improve a model's accuracy.

Finally, due to this limitation, I feel that future research should focus on improving the uses for these classifiers rather than continuing to improve model accuracy. As I discussed in Section 1.1, the knowledge of someone's income is extremely important and powerful. Therefore, it is essential to handle that power in the correct ways.

7 | REFERENCES

[1] Matta, Christy. "How Does Your Income Determine Your Well-Being?" *Mental Help How Does Your Income Determine Your WellBeing Comments*, https://www.mentalhelp.net/blogs/how-does-your-income-determine-your-well-being/.

[2] Bureau, US Census. "Why We Ask about... Income." *Why We Ask About... Income | American Community Survey | US Census Bureau*, https://www.census.gov/acs/www/about/why-we-ask-each-question/income/.

[3] "Facebook Introduces Household Income Targeting Based on U.S. ZIP Code Averages." *MarTech*, 25 Oct. 2021, https://martech.org/facebook-introduces-household-income-targeting-based-on-u-s-zip-code-averages/.

[4] Singh, Author Abhinav. "Predicting Income Level, an Analytics Casestudy in R." *CloudxLab Blog*, 13 Sept. 2017, https://cloudxlab.com/blog/predicting-income-level-case-study-r/.

[5] youngcr003. "Predicting Income Using US Census Data." *Kaggle*, Kaggle, 15 Sept. 2017, https://www.kaggle.com/youngcr003/predicting-income-using-us-census-data.

[6] "Income Prediction." *Income Prediction*, https://seahorse.deepsense.ai/casestudies/income_predicting.html.

[7] Datopian. "Adult Income." *DataHub*, https://datahub.io/machine-learning/adult#data.

[8] *Barry Becker - General Manager - US ... - Linkedin.cn*. https://www.linkedin.cn/in/barry-becker-25a2718.

[9] Bureau, US Census. "Combining Data – a General Overview." *Census.gov*, 18 Nov. 2021, https://www.census.gov/about/what/admin-data.html.

[10] Lee, Corliss. "Who, How Many and Where: Research Using the U.S. Census." *UC Berkeley Library Update*, 8 Oct. 2016, https://update.lib.berkeley.edu/2016/05/04/who-how-many-and-where-research-using-the-u-s-census/.