A mathematical essay on naive bayes classifier

Gautham Govind A
Dept. of Electrical Engineering
Indian Institute of Technology Madras
ee19b022@smail.iitm.ac.in

***Abstract*—The objective of this assignment is to explore the mathematical formalism behind naive bayes classifier and then to use it in a real-life application. In this assignment, as a real-life application, naive bayes classifier is used to formally identify the factors which could have been used to predict the income category of an adult, using the 1994 US census data. Data visualization, cleaning and modelling is done using Python. The analysis enables us to arrive at the conclusion that it is possible to make reasonable predictions regarding the income of an adult using factors including but not limited to education, working class and gender. This is a reworked version of the original assignment with improvements to the Modelling section in the form of enhanced benchmarking of the model including a comparison with an SVM model.**

***Index Terms*—naive bayes, python, visualization, predictive modelling, binary classification**

I. Introduction

Given a set of features and a target variable, predictive modelling is typically used for generating a model which can make predictions for cases where we do not know the value of the target variable, i.e., only the features are available. Apart from this use, a model can also be used for developing an intuition of how various factors influence the target variable. In this assignment, we try to make use of a model for the purpose of identifying the key factors which influence the decision in a classification problem.

In particular, we make use of Naive Bayes classifier for the purpose of identifying relationships in a classification problem. Naive bayes classifiers are a family of simple probabilistic classifiers based on applying Bayes' theorem with strong independence assumptions between the features. Though they are by nature very simple, they can give good results in many problem settings. Naive bayes is a very flexible classifier in the sense that it can accommodate both continuous and categorical variables. We will make use of this flexibility for designing a robust model which can solve a classification problem.

In our problem setting, the goal is to use naive bayes classifier to predict the income category of an individual given a variety of factors like education, working class, gender etc. We make use of the publicly available 1994 US census data for building the model. After building the model, we evaluate the model using a variety of evaluation metrics. By examining how well the model performs, we can identify how good the identified relationships are.

Section II gives an overview of the various techniques used for data cleaning, data visualization and an initial exploratory analysis. A lot of insights can be gained just by making qualitative observations from the given data. Section III gives a short description of the mathematical formalism behind naive bayes. Section IV describes the various models that were tried and the results that were obtained by applying naive bayes in this particular case. Section V gives a summary of the major conclusions drawn from the analysis.

II. Exploratory Data Analysis

In this section, we describe the process of data cleaning and data visualization. We also make some qualitative observations.

A. Preliminary analysis

The given dataset has 32561 rows and 15 columns. It must be noted that the dataset itself lacked any column headings: they had to be added in manually. A brief overview of the dataset is presented in Figure 1. We observe that we have both categorical as well as numerical variables. It also seems that there are no null values. From the common statistics (mean, median, mode) of the numerical variables, we make the following observations:

- The distribution has representatives in the age group 17 - 90. This ensures that we capture the variation across different age groups.
- Every individual has had at least 1 year of education. Note that this might not have been the case had the census been done in a developing country instead of the US.
- Capital gain and loss values are 0 each for a majority of the population, since the median is 0. This could be because a large section of the population is not involved in capital asset transactions, which is rather intuitive.

We also look at the distribution for our target class, which is income. On plotting, we obtain Figure 2. We have two categories: income $\leq$ 50K (low income category) and income $>$ 50K (high income category). Of course for income, low and high may be subjective, but for our purposes we go ahead with this nomenclature. The count of individuals in the high income category is only about a third of the count of individuals in the low income category. We will have to account for this during model building.

B. Feature by feature analysis

For each feature, we perform the following analyses:

- For continuous features, we first explore how the feature itself is distributed. For Gaussian Naive Bayes to be

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 32561 entries, 0 to 32560
Data columns (total 15 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   age             32561 non-null  int64
 1   workclass       32561 non-null  object
 2   fnlwgt          32561 non-null  int64
 3   education       32561 non-null  object
 4   education-num   32561 non-null  int64
 5   marital-status  32561 non-null  object
 6   occupation      32561 non-null  object
 7   relationship    32561 non-null  object
 8   race            32561 non-null  object
 9   sex             32561 non-null  object
 10  capital-gain    32561 non-null  int64
 11  capital-loss    32561 non-null  int64
 12  hours-per-week  32561 non-null  int64
 13  native-country  32561 non-null  object
 14  income          32561 non-null  object
dtypes: int64(6), object(9)
memory usage: 3.7+ MB
```

Fig. 1. Summary of the dataset

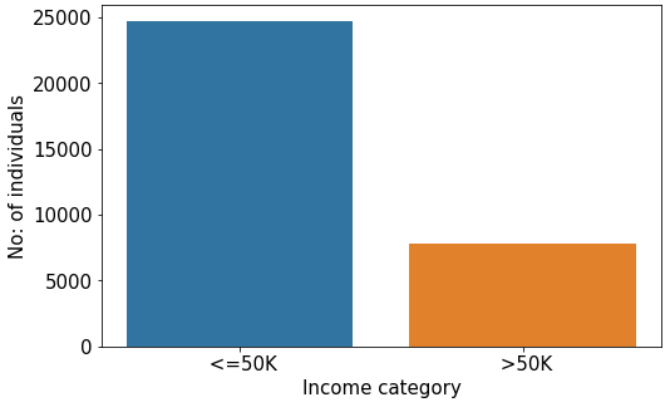

Fig. 2. Target class imbalance

effective an important requirement is to have the variable distributed approximately normally.

- Then we make appropriate plots for continuous and categorical features to explore if we can derive any insights from a visual standpoint. Only features which provide insights are reported though all features were tested.

Continuous variables

We make use of kernel density estimate (kde) plots for examining the distribution of continuous variables. KDE plots are essentially smoothed histograms. Since they are less cluttered versions of histogram plots, insights are more obvious.

For applying naive bayes, it is essential to ensure that none of the continuous features are correlated. The correlation matrix is shown in Figure 3. **Clearly, the correlation values are very small, thereby validating the naive bayes assumption of independence of features.**

We explore the continuous variables one by one below:

Age

We obtain the plot as given in Figure 4. Although the plot does resemble a Gaussian, it is skewed. In an attempt

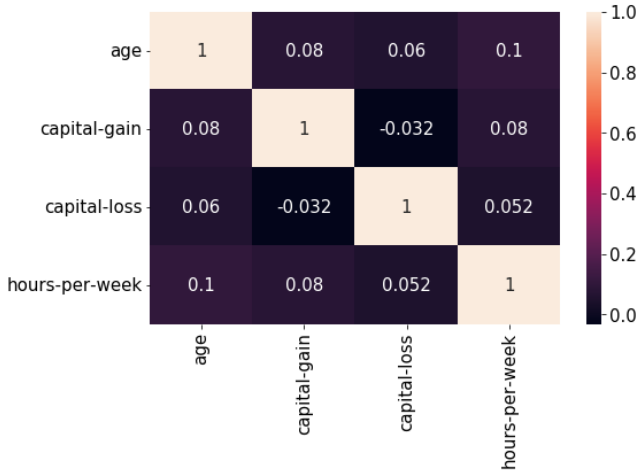

Fig. 3. Correlation for continuous features

to make it more like a Gaussian, we can apply the box-cox transformation. It transforms a datapoint y as follows:

$$y(\lambda) = \begin{cases} \frac{y^\lambda - 1}{\lambda}, & \text{if } \lambda \neq 0 \\ \log y, & \text{if } \lambda = 0 \end{cases}$$

λ is treated as a hyper-parameter and the most suitable value is chosen. After performing the box-cox transformation, the age variable distribution gets transformed and the transformed feature is illustrated in Figure 5. Clearly, the distribution more closely resembles a Gaussian now. We will use this transformed quantity while applying Gaussian Naive Bayes.

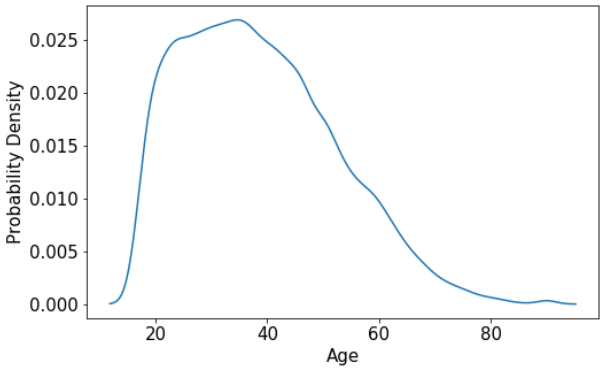

Fig. 4. Density plot for Age

On plotting the distribution of age according to income category, we obtain the plot in Figure 6. We observe that:

- The **peak for income group $\leq$ 50K occurs close to the age 20.** This is intuitive as we expect people in their 20s to make less income as they are probably just starting out on jobs/ businesses.
- **The peak for income $>$ 50K occurs close to the age 40.** This is also intuitive as we expect most people to be at their peak earning capacity around this age. As they grow

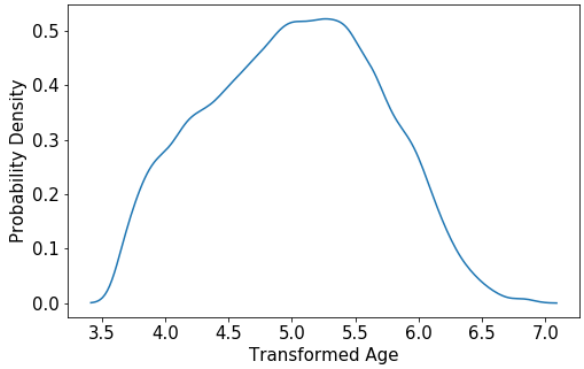

Fig. 5. Density plot for transformed age variable

older, they move closer to retirement, possibly adversely affecting their income.

- We notice that there is a **greater proportion of people with income $\leq$ 50K as compared to $>$ 50K for any given age,** which is what we would expect from any society.

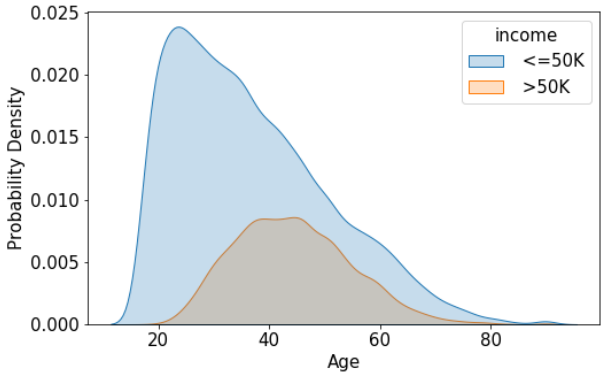

Fig. 6. Age distribution for different income categories

fnlwgt

No description of this variable was given in the problem statement. After scouring the internet to find what this variable actually represents, the following description was found in the official dataset description:

"People with similar demographic characteristics should have similar weights. There is one important caveat to remember about this statement. That is that since the CPS sample is actually a collection of 51 state samples, each with its own probability of selection, the statement only applies within state."

However, we clearly have no information regarding which state an individual is from. **Hence, for our purposes, we can safely discard this variable** as this variable does not add any value, as long as we don't have any information regarding state. For the sake of completeness, the distribution of this variable is given in Figure 7.

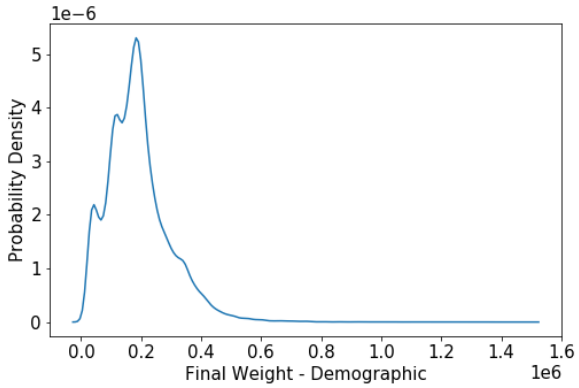

Fig. 7. Distribution of the variable fnlwgt

Years of education

We obtain the distribution plot as given in Figure 8. We see that the distribution doesn't resemble a Gaussian at all. We also note that whatever information is captured by years of education is also captured by the 'education' column, which is categorical and hence is easier to handle with naive bayes. So we drop this column and do not use it for Naive Bayes model.

On plotting the distribution of years of education according to income category, we obtain the plot in Figure 9. We observe that in general, **people who earn more tend to have invested a longer duration in education.**

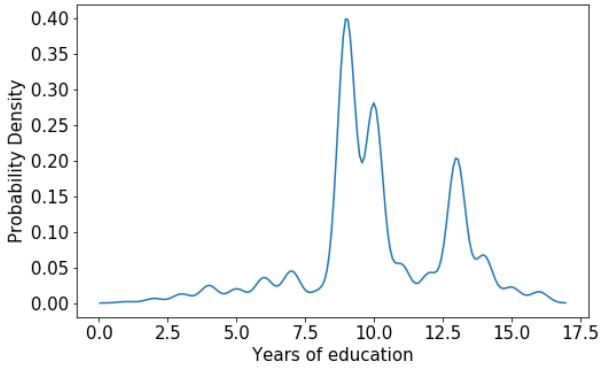

Fig. 8. Distribution of the number of years of education

Capital gain

We obtain the distribution plot as given in Figure 10. Although the right tail is longer than left tail, since the values are almost zero along the right tail, we consider this to be a Gaussian and hence do not apply any transform.

On plotting the distribution of capital gain according to income category, we obtain the plot in Figure 11. Capital

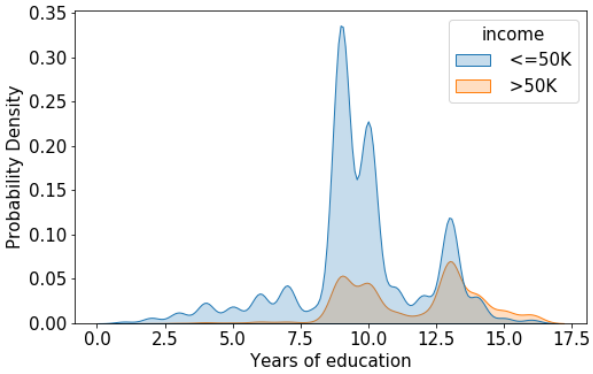

Fig. 9. Number of years of education for different income categories

gain plots don't seem to reveal any information regarding the dataset. This could be because more than 50% of the points have the capital gain value as 0 as a majority of the population do not involve in capital asset transactions. So we try plotting only the non-zero values to see if we can infer anything. We obtain the plot given in Figure 12. Clearly, we observe that a **higher capital gain has a higher probability density for people in the high income bracket,** which makes sense intuitively.

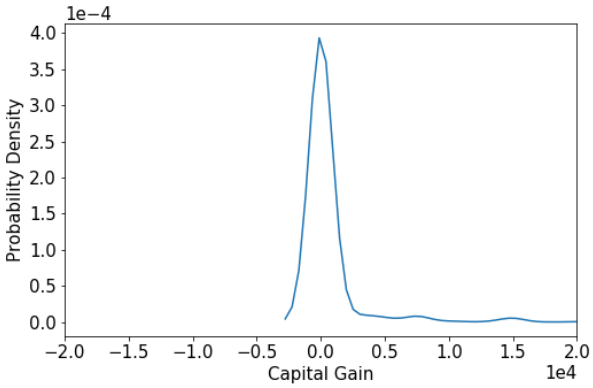

Fig. 10. Distribution of the capital gain

Capital loss

We obtain the distribution plot as given in Figure 13. Again the right tail is longer than left tail, however since the values are almost zero along the right tail, we consider this to be a Gaussian and hence do not apply any transform.

Since the situation is same as that of capital gain, so we try plotting only the non-zero values to see if we can infer anything. We obtain the plot given in Figure 14. Counter-intuitively, it seems like people with higher income have higher capital losses on average! However, this could be because the more you get involved in capital asset transactions, the more are your chances for gain/loss. Individuals could still be in the high income category if the gains offset losses.

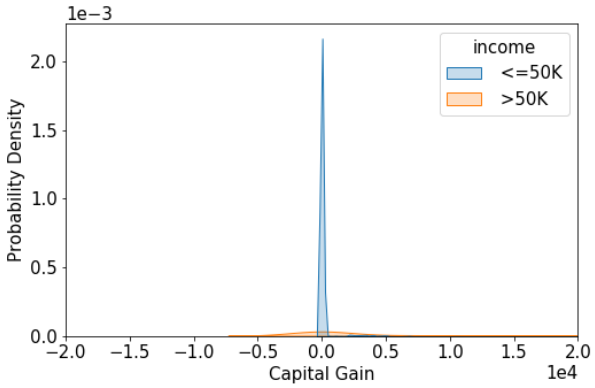

Fig. 11. Capital gain for different income categories

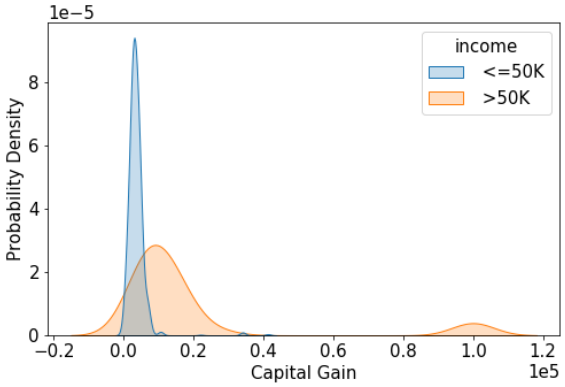

Fig. 12. Capital gain considering only non-zero values

Hours per week

We obtain the distribution plot as given in Figure 15. The distribution is sufficiently close to Gaussian, so we do not use any transforms.

On plotting the distribution of working hours per week according to income category, we obtain the plot in Figure 16. We make the following observations:

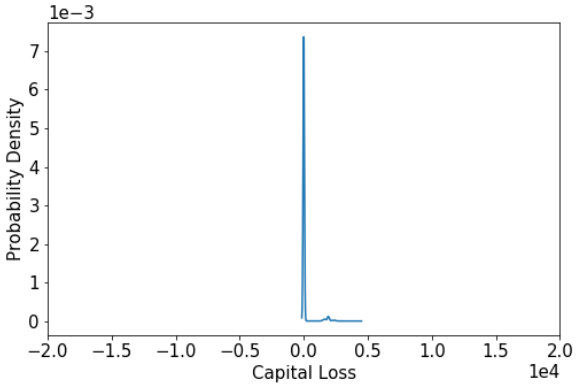

Fig. 13. Distribution of the capital loss

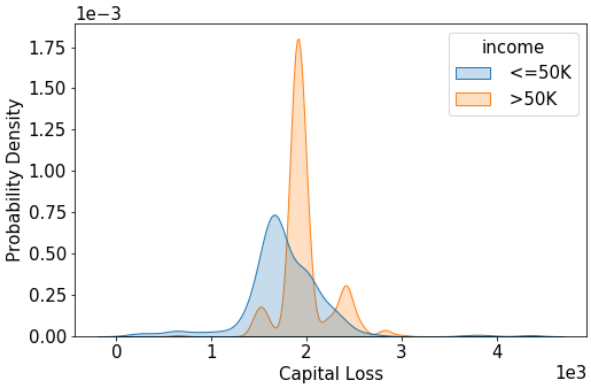

Fig. 14. Capital loss considering only non-zero values

- Contrary to what one would expect, it appears that working more hours does not necessarily mean your income will be higher.
- However, it can be seen that a higher proportion of high income population works for more than 40 hours as compared to the lower income population.

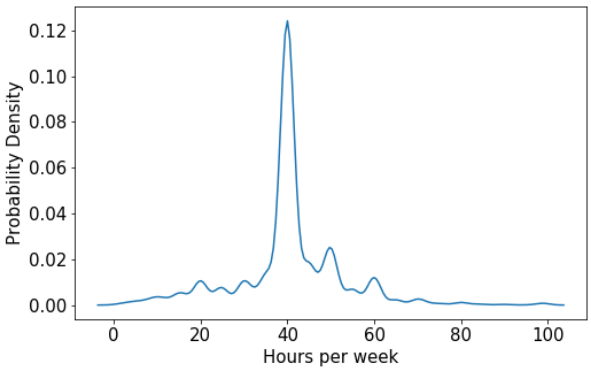

Fig. 15. Distribution of the working hours per week

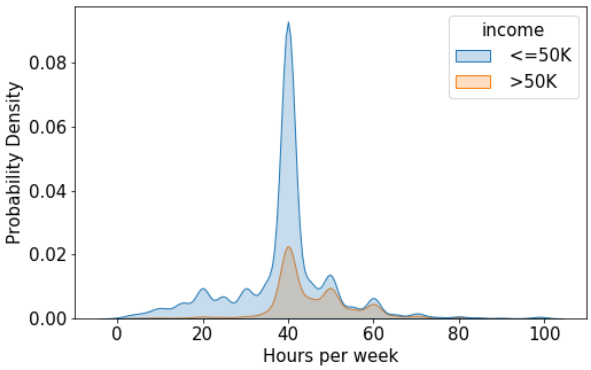

Fig. 16. Working hours per week for different income categories

Categorical variables

For the case of categorical variables, we make use of count plots instead of kde plots for visualization. We plot the count of each category of a categorical variable according to the two income categories.

Working class

We obtain the count plot as shown in Figure 17. We make the following observations:

- Self-employed(inc) is the only category with more individuals belonging to high income category as compared to low income category. This shows that you are more **likely to earn better if you manage to start a successful business for yourself.**
- Private sector has a high proportion of people from both high-income and low-income categories. This evidences the infamous **income disparity in the private sector.**
- There are some individuals for which the working class is denoted by ?'. These represent individuals for who we do not know the working class.

On examination, it is found that only 5.6% of the total number of entries have the working class missing. Since the number is marginal, we drop these rows.

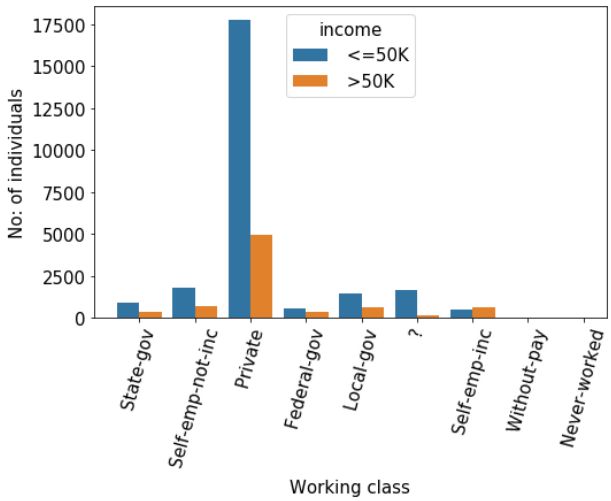

Fig. 17. Distribution of working class

Education

We obtain the count plot as shown in Figure 18. We make the following observations:

- Although we cannot make a blanket statement that education directly ensures high income, we see that the **only categories with proportion of high income group more than proportion of low income group are Masters and Doctorate groups.**
- The proportion of high income individuals is very low for people who did not attend university, again implying that **education has a significant role in determining income.**

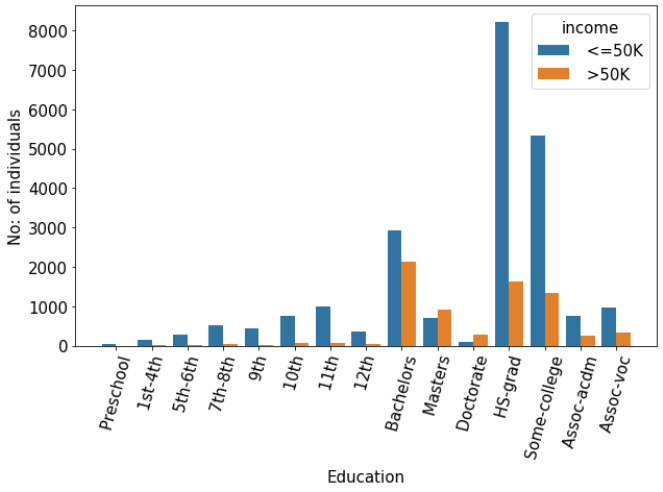

Fig. 18. Distribution of education levels

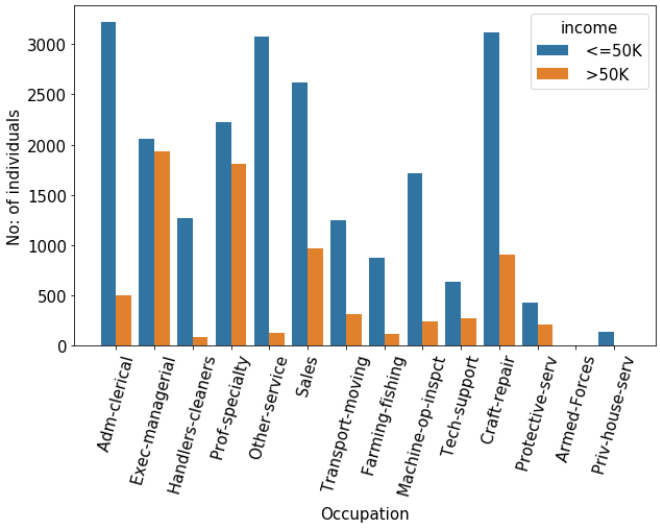

Fig. 20. Distribution of occupation

Marital status

We obtain the count plot as shown in Figure 19. We observe that Married-civ-spouse is the only category which has comparable number of people belonging to both categories of income. For other categories, the fraction of people belonging to high income category is marginal. This might be because married individuals may be more concerned about their financial security.

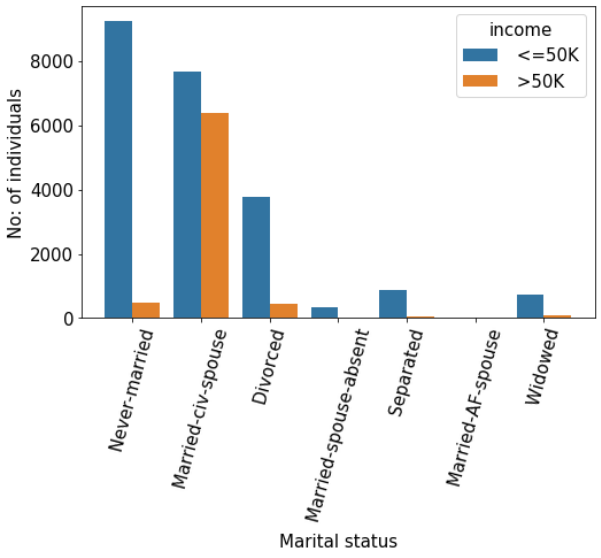

Fig. 19. Distribution of marital status

Occupation

We obtain the count plot as shown in Figure 20. We observe that **managerial executives and speciality professionals are likely to earn more.** This also makes sense intuitively as these are often considered as the "high-paying" professions.

Gender

We obtain the count plot as shown in Figure 21. We observe that **on an average, males have more chance to earn more than females,** and that about a third of males earn more than 50k income.

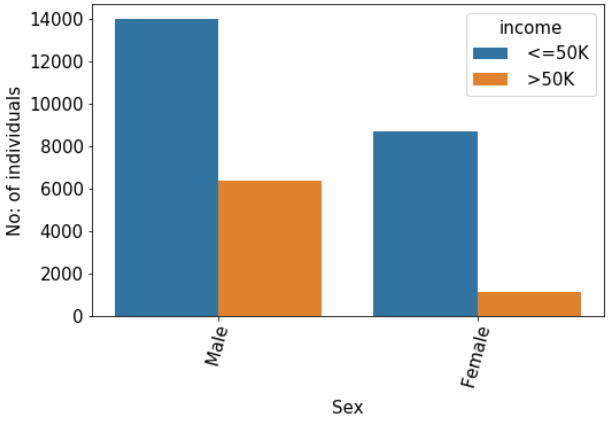

Fig. 21. Distribution of gender/sex

III. Model: Naive Bayes Classifier

In this section, we will give a brief overview of the mathematical formalism behind the naive bayes classifier.

Naive Bayes classifiers are a family of simple "probabilistic classifiers" based on applying Bayes' theorem with **strong (naive) independence assumptions between the features.**Naive Bayes classifiers are **highly scalable**, requiring a number of parameters linear in the number of variables/features in a learning problem. Maximum-likelihood training, which is the most generally used strategy, can be done by **evaluating a closed-form expression** which takes linear time, rather than by expensive iterative approximation, such as gradient descent, as used for many other types of

classifiers. Despite the seemingly strong assumptions, naive bayes classifiers have been shown to work well in a lot of real-life scenarios.

A. Probabilistic Model

In any classification problem, our goal is to find the class C_k in which a given datapoint **x**, which will be an n-dimensional vector, where n is the number of features, belongs. Then, more formally, our goal is to find $P(C_k|\mathbf{x})$ for each class C_k.

Using Bayes' theorem,

$$P(C_k|\mathbf{x}) = \frac{P(C_k, \mathbf{x})}{P(\mathbf{x})}$$

For a given datapoint, the denominator is fixed, implying that the class to which the point should be assigned is influenced solely by the numerator which essentially contains the joint probability function $P(C_k, \mathbf{x})$. Now,

$$P(C_k, \mathbf{x}) = P(C_k, x_1, x_2, ..., x_n)$$

where $x_1, x_2..., x_n$ are the n features. **This is where we make use of the strong naive bayes assumption of mutual independence of features, conditional on the class** C_k Using this assumption,

$$P(C_k, x_1, x_2, ..., x_n) = P(C_k)P(x_1|C_k)P(x_2|C_k)...P(x_n|C_k)$$

Therefore, we can write:

$$P(C_k|\mathbf{x}) \propto P(C_k)\Pi_{i=1}^{n}P(x_i|C_k)$$

$P(C_k)$ is termed the **prior probability** for class C_k, since this can be interpreted as the probability of observing class C_k before any observations were made. If we know the distribution of the classes before measurements through some **domain knowledge, this can be used for assigning the prior probabilities.** If not, we can always choose a uniform prior which would mean assigning equal probabilities to all classes.

By making the simplifying assumption of independence of features, we are required to compute only the distributions $P(x_i|C_k)$ for $i \in \{1, 2, ..., n\}$, instead of the joint distribution. How does this make the computations easier?

To see this, let us assume all distributions are normal. If we do not assume conditional independence, we will be required to compute $P(x_1, x_2, ..., x_n|C_k)$ which essentially is a joint Gaussian having n variables. Clearly, the covariance matrix of such a Gaussian will n^2 entries and hence the estimation of parameters will be $O(n^2)$. However, if we assume independence, this can be broken down into n separate Gaussians with just 1 variable each. Clearly, such a model would only require $O(n)$ parameter estimations, giving us a **quadratic speedup**. This can immensely help in computations, especially while dealing with reasonably large datasets.

Notice that this simplifying assumption reduces the complexity of the model. This makes naive bayes a **high bias, low variance classifier.**

Typically, for continuous features we assume the distribution to be Gaussian and for categorical variables, we assume the distribution to be bernoulli/categorical distribution. After choosing a distribution, we estimate the parameters using Maximum Likelihood Estimation.

B. MAP decision rule

After obtaining the probabilities $P(C_k|\mathbf{x})$, we must make a prediction to assign a class. The most commonly used prediction strategy is to **pick the class with the highest value** of $P(C_k|\mathbf{x})$. This is termed as the maximum a posteriori or MAP decision rule.

C. Gaussian Naive Bayes

Gaussian Naive Bayes is a typically used model for continuous variables. We essentially assume the Gaussian distribution:

$$f_x(x|C_k) = \frac{1}{\sqrt{2\pi}\sigma_k}e^{\frac{-(x-\mu_k)^2}{2\sigma_k^2}}$$

for the probability density function for class C_k.

We estimate μ_k, σ_k and the prior $\pi_k(P(C_k))$for each class C_k as follows:

$$\hat{\mu}_k = \frac{\sum_{i\forall n_k} x_i}{n_k}$$

$$\hat{\sigma}_k^2 = \frac{\sum_{i\forall n_k}(x_i - \hat{\mu}_k)^2}{n_k - 1}$$

$$\hat{\pi}_k = \frac{n_k}{n}$$

where n_k is the number of observations of the k$^{\text{th}}$ class and n is the total number of observations.

If we assume shared variances across the different classes, we will obtain **linear decision boundaries.**

D. Categorical Naive Bayes

When the input feature is categorically distributed, we use categorical naive bayes. In categorical naive bayes, we assume the probability distribution for each feature to be a categorical variable. Suppose a featurex_j can take values in categories $1, 2, ..., T$. Then mathematically, the probability for feature x_j to take value as category t, given a class C_k is estimated as:

$$P(x_j = t|C_k) = \frac{N_{kt}}{N_k}$$

where N_{kt} is the number of samples in class C_k which have category of x_j as t and N_k is the total number of samples in class C_k. We then calculate the probability of observing the sample X_i given a class C_k by multiplying out the above calculated probabilities for each feature.

However, this approach has one drawback: if we encounter a category which we have never seen before, the entire product vanishes because the probability for this particular category goes to zero. To prevent this, a smoothing parameter α, which is typically a small value greater than zero, is added to both the numerator and denominator:

TABLE I
METRICS FOR MODEL 1 (UNIFORM PRIOR)

Metric	Score
Accuracy	0.831
Precision	0.692
Recall	0.578
F1 Score	0.630

$$P(x_j = t|C_k) = \frac{N_{kt} + \alpha}{N_k + \alpha}$$

IV. MODELLING

In this section, we discuss the application of the naive bayes classifier to our problem.

We create and evaluate two separate models:

1) In Model 1, we **set the prior uniformly for both classes,** i.e., we assign 0.5 each for the prior values.
2) In Model 2, we **set the prior according to the class distribution,** i.e we set 0.751 for the low-income category and 0.249 for the high-income category.

Each model is generated using a **"mixed" naive bayes approach.** This is because we have both numerical and categorical variables. So we fit a Gaussian naive bayes classifier on the continuous features, a categorical naive bayes classifier on the categorical features, get the output probabilities and multiply them out to get the total probabilities from the model.

We evaluate the models based on multiple metrics. A short description of the used metrics is given below:

- **Accuracy:** Accuracy is simply the **ratio of number of correct predictions to total number of predictions.** Although this seems like a very good metric intuitively, accuracy fails on classification problems with a skewed class distribution because of the intuitions developed by practitioners on datasets with an equal class distribution.
- **Precision:** Precision is the **ratio of true positives to the total positive predictions.** Precision is typically used when the cost of false positive is high. For instance, email spam detection.
- **Recall:** Precision is the **ratio of true positives to the total positive ground truths.** Recall is typically used when the cost of false negative is high. For instance, in fraud detection or sick patient detection.
- **F1-score:** F1-score is simply a **harmonic average of precision and recall.** F1 Score is typically used if we need to seek a balance between Precision and Recall and there is an uneven class distribution.

A. Model 1

The confusion matrix for model 1 is shown in Figure 22. The values for the evaluation metrics are given in Table I.

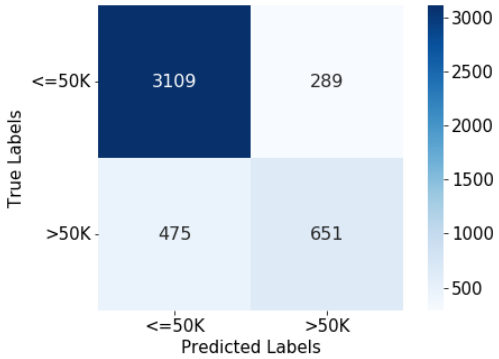

Fig. 22. Confusion matrix for model 1 (uniform prior)

TABLE II
METRICS FOR MODEL 2 (NON-UNIFORM PRIOR)

Metric	Score
Accuracy	0.811
Precision	0.746
Recall	0.363
F1 Score	0.489

B. Model 2

The confusion matrix for model 2, using prior based on class distribution, is shown in Figure 23. The values for the evaluation metrics are given in Table II.

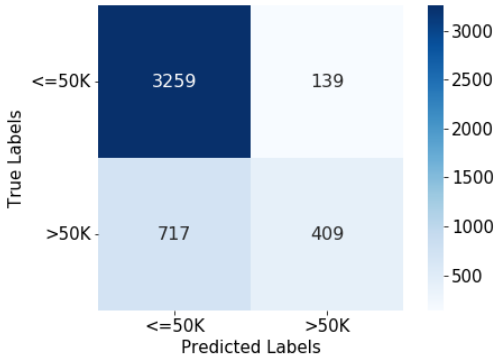

Fig. 23. Confusion matrix for model 2 (non-uniform prior)

We observe that **Model 2 has better precision whereas Model 1 performs better in all other metrics.** Although we might consider Model 2 in settings where precision is of utmost importance, in this case since Model 1 does better in all other aspects, we choose **Model 1 to be the better model.**

ROC curve is another common tool used with binary classifiers. A Receiver Operating Characteristic curve, or ROC curve, is a graphical plot that illustrates the diagnostic ability of a binary classifier system as its discrimination threshold is varied. The ROC curve is created by plotting the true positive rate (TPR) against the false positive rate (FPR) at various

TABLE III
METRICS FOR GAUSSIAN NAIVE BAYES

Metric	Score
Accuracy	0.787
Precision	0.650
Recall	0.313
F1 Score	0.423

threshold settings. The ROC curve for Model 1 is given in Figure 24.

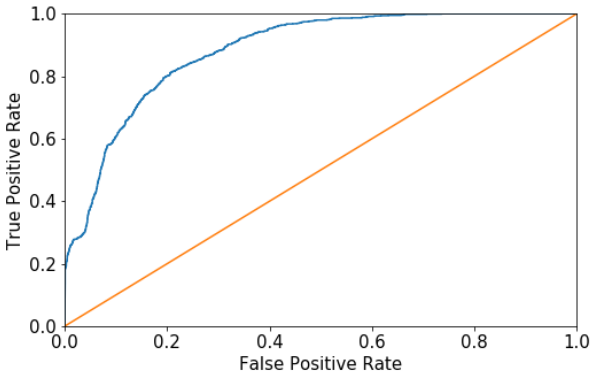

Fig. 24. ROC curve for Model 1

The dotted line represents the ROC curve of a purely random classifier; a good classifier stays as far away from that line as possible (toward the top-left corner). One way to compare classifiers is to measure the area under the curve (AUC). A perfect classifier will have a ROC AUC equal to 1, whereas a purely random classifier will have a ROC AUC equal to 0.5. **The area under the ROC curve of our naive bayes classifier is 0.882.**

To get a sense of how much continuous variables and categorical variables contribute to the prediction quality, we take the Gaussian naive bayes and categorical naive bayes (with uniform prior) separately and make predictions. For Gaussian naive bayes, we obtain the confusion matrix shown in Figure 25 and metrics shown in Table III. For categorical naive bayes, we obtain the confusion matrix shown in Figure 26 and metrics shown in Table IV.

From the confusion matrix as well as the metrics, it is clear that the Gaussian naive bayes contributes mostly to the precision metric whereas the categorical naive bayes dominates all other metrics. If we were to choose between the two, **categorical naive bayes is more relevant as it leads to a more balanced model. Hence, we can conclude that the categorical variables contribute more to the income category prediction.**

Comparison with SVM

Despite trying multiple configurations for the naive bayes model, the model still has rather low values for the evaluation metrics. This is probably because of the underlying naive

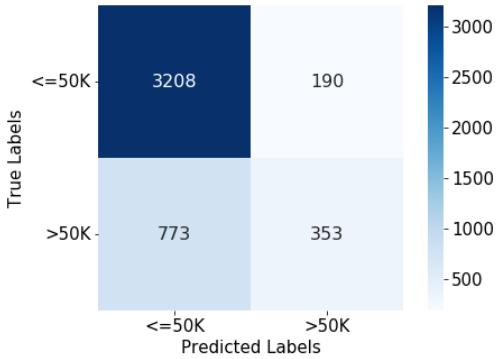

Fig. 25. Confusion matrix for Gaussian naive bayes

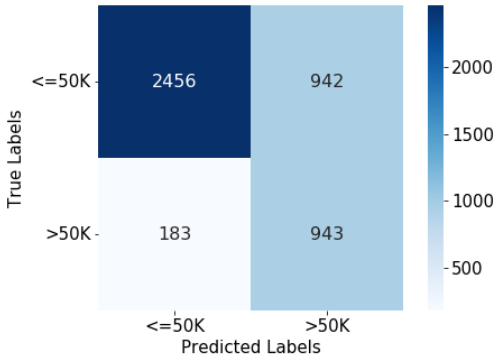

Fig. 26. Confusion matrix for categorical naive bayes

assumptions of the model. To see if this is indeed the case, we build a support vector machine (SVM) model on the same dataset. Since SVM does not have naive independence assumptions, an enhancement of performance would imply that it was indeed these assumptions which hampered the performance of the naive bayes model.

We create multiple SVM models based on values of different hyperparameters. We use the gaussian kernel and grid search on the hyperparameter space. The best model which we obtain produces a confusion matrix as shown in Figure 27. The evaluation metrics are given in Table V.

It is evident that the SVM model outperforms the naive bayes model in terms of F1-score, which is our metric of choice. It is to be noted that the SVC classifier takes much longer on average to train as compared to naive bayes. This

TABLE IV
METRICS FOR CATEGORICAL NAIVE BAYES

Metric	Score
Accuracy	0.751
Precision	0.500
Recall	0.837
F1 Score	0.626

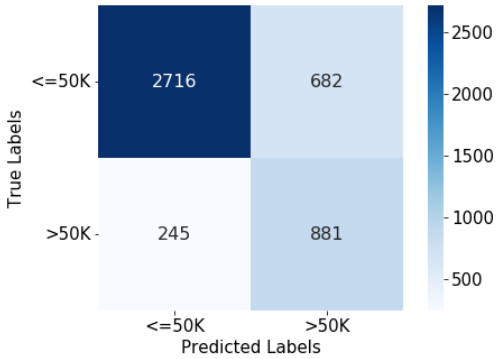

Fig. 27. Confusion matrix for SVM

TABLE V
METRICS FOR SVM

Metric	Score
Accuracy	0.795
Precision	0.564
Recall	0.782
F1 Score	0.655

is to conclude that even though SVM performs better, naive bayes does possess discriminatory characteristics to separate out the classes and hence would be useful in setting a minimum benchmark in large datasets.

V. CONCLUSIONS

From our extensive analysis of the given dataset using naive bayes calssifier, we arrive at the following conclusions:

- Naive bayes classifier, despite having rather strong assumptions, performs reasonably well as long as the features are more or less independent and the continuous variables are approximately Gaussian.
- Changing the prior can significantly affect the performance of the naive bayes model as evidenced by the improvement in performance using a uniform prior instead of class weighted prior.
- Categorical features provide more insights regarding the value of target variable as compared to continuous features as is evident from the fact that the categorical naive bayes classifier outperforms the Gaussian naive bayes classifier.
- The income bracket of an individual can be predicted to a reasonable extent using factors like education, age, profession and gender.
- While it is acceptable that education and working hours influence the income, it must be critically examined if certain groups based on gender and race face barriers while aspiring to be in the high income category to ensure wellness and propserity of the society as a whole.

VI. AVENUES FOR FURTHER RESEARCH

Naive bayes classifier, despite working reasonably well, has very strong assumptions which might not be warranted in many cases. Using other models like SVM which do not require such strong assumptions might lead to better results. Further, rather than using Gaussian naive bayes, if we can more accurately model the distribution of continuous variables, we might be able to achieve better performance.

REFERENCES

[1] A. Geron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. O'Reilly Media, Inc., 2nd ed., 2019.
[2] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
[3] J. D. Hunter, "Matplotlib: A 2d graphics environment," *Computing in Science & Engineering*, vol. 9, no. 3, pp. 90–95, 2007.
[4] Wes McKinney, "Data Structures for Statistical Computing in Python," in *Proceedings of the 9th Python in Science Conference* (Stéfan van der Walt and Jarrod Millman, eds.), pp. 56 – 61, 2010.
[5] "Naive bayes classifier." https://en.wikipedia.org/wiki/Naive_Bayes_classifier.