**Project Report**

**CSC-869 Data Mining**

# Mini Project 1: Naïve Bayesian Classifier

Submitted to

Prof. Hui Yang

By-

Akshay Kasar

SFSU ID: 918812874

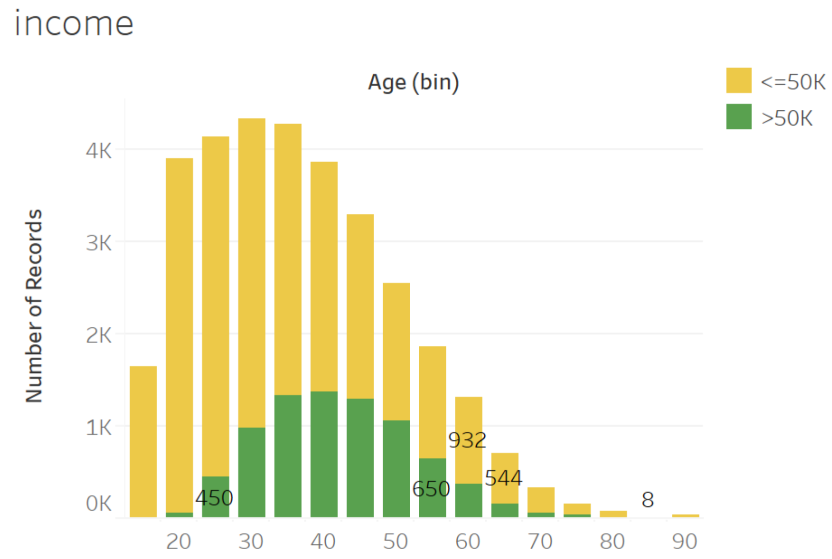
## Problem Statement:

Predict whether income exceeds $50K/yr based on census data. Also known as "Census Income" dataset.

## Visualizations:

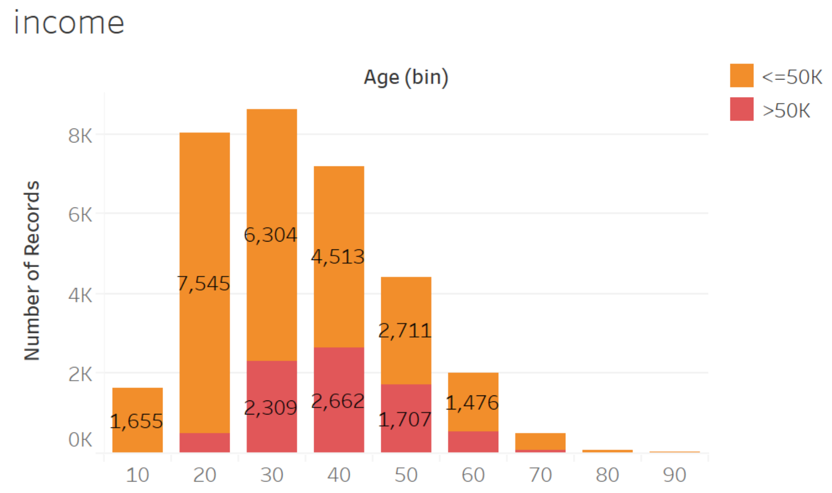
1) Age Group vs Income Distribution Bargraph

a. Bin size = 5



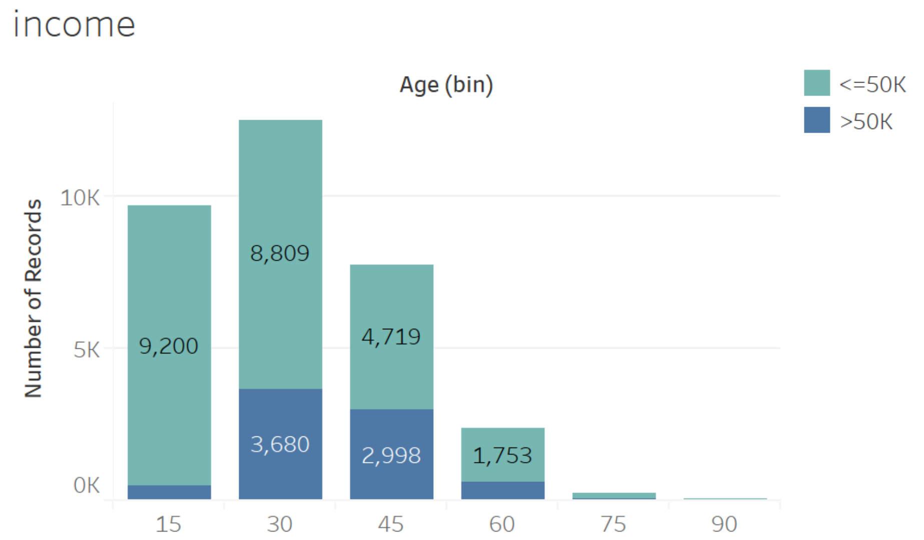
Sum of Number of Records for each Age (bin). Color shows details about Income. The marks are labeled by sum of Number of Records.

b. Bin size = 10



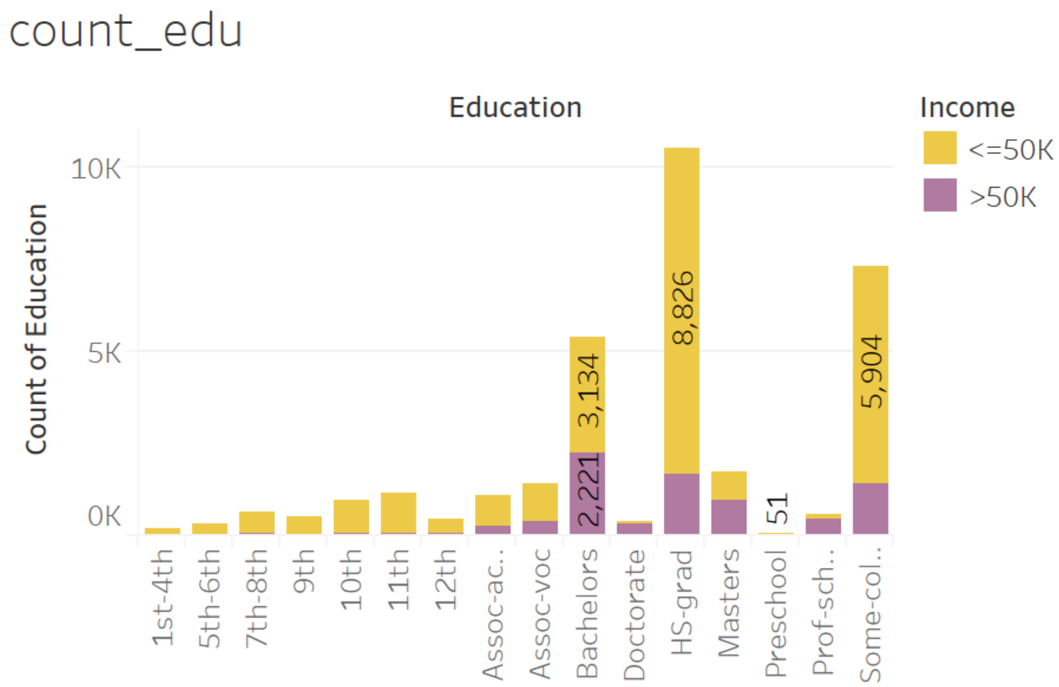
Sum of Number of Records for each Age (bin). Color shows details about Income. The marks are labeled by sum of Number of Records.

c. Bin size = 15



Sum of Number of Records for each Age (bin). Color shows details about Income. The marks are labeled by sum of Number of Records.

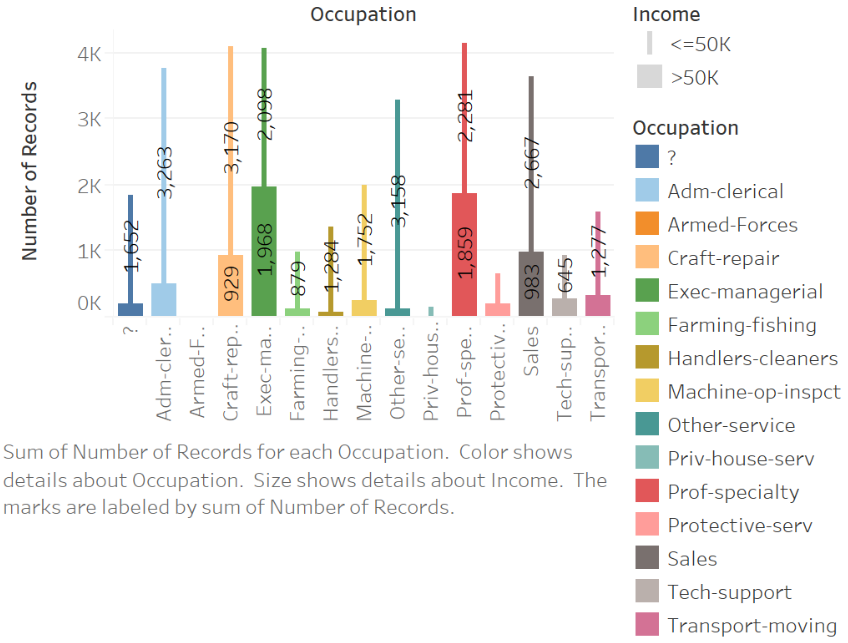
2) Education vs income



Count of Education for each Education. Color shows details about Income.

## 3) Hours-per-week vs income

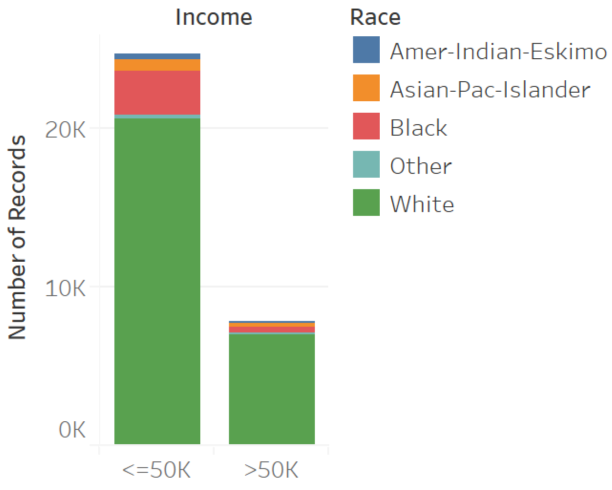
sex_occupation_income



Sum of Number of Records for each Occupation. Color shows details about Occupation. Size shows details about Income. The marks are labeled by sum of Number of Records.

## 4) Race_income

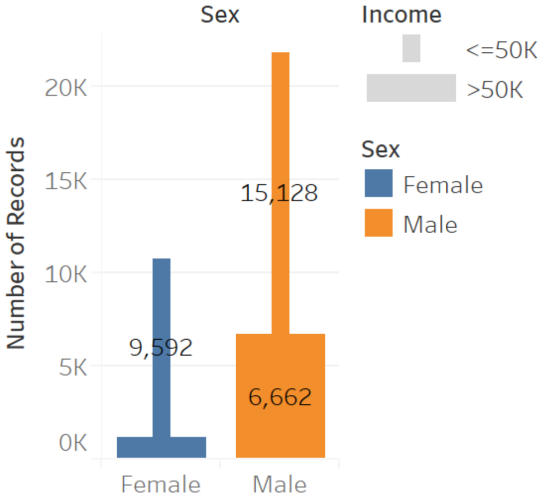
race_income



Sum of Number of Records for each Income. Color shows details about Race.

## 5) Gender vs income

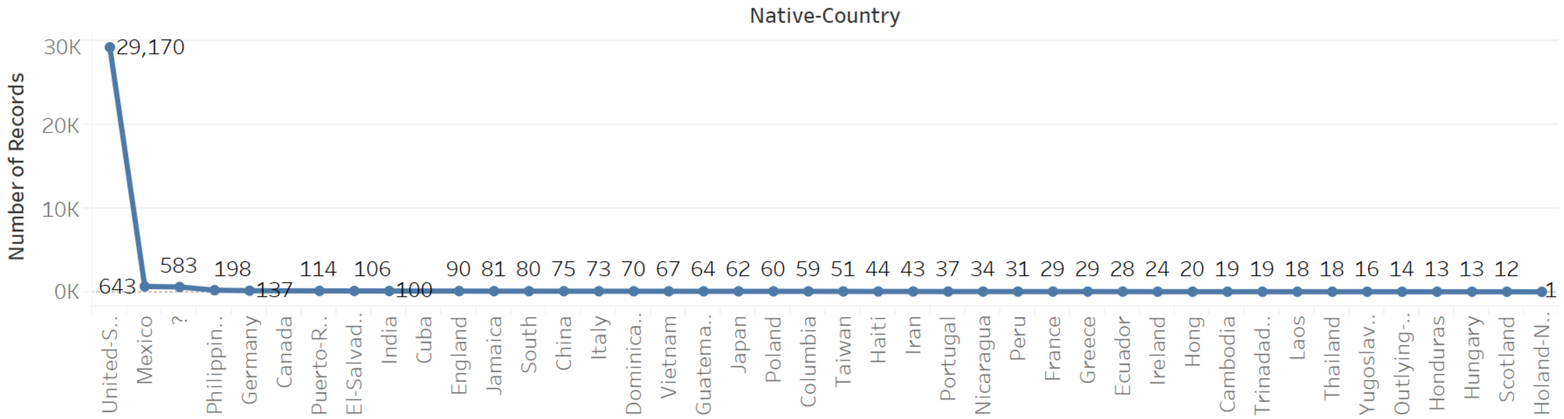
sex_occupation_income



Sum of Number of Records for each Sex. Color shows details about Sex. Size shows details about Income. The marks are labeled by sum of Number of Records.

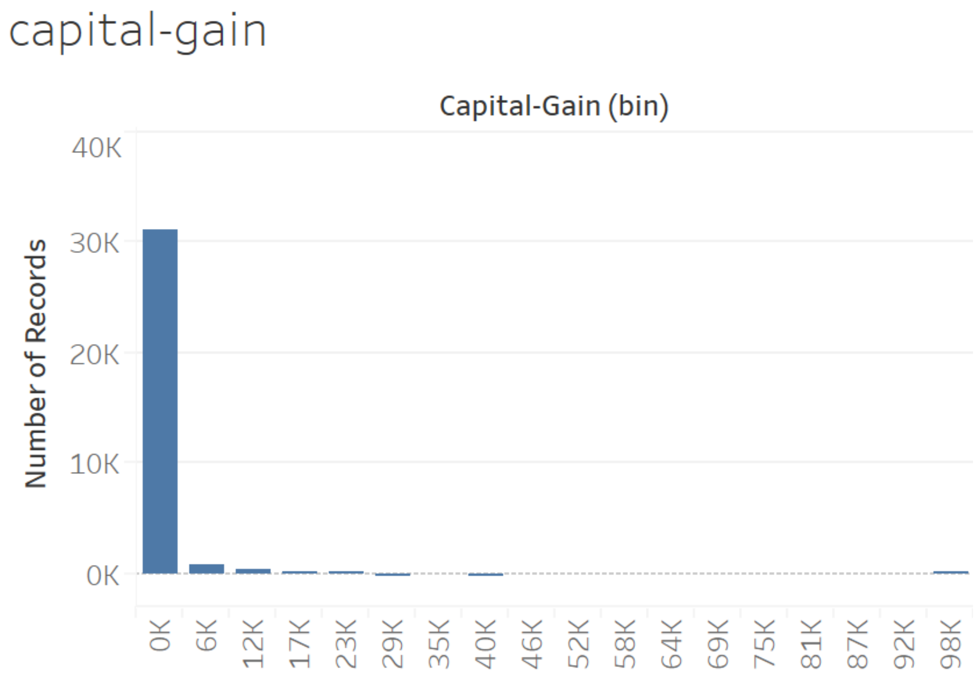
## 6) native_country_distribution

native_country



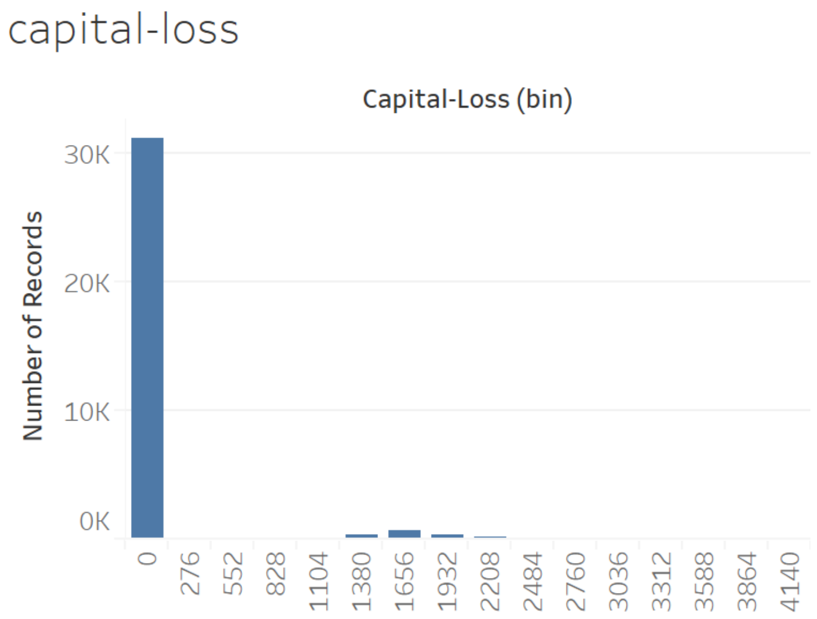
The trend of sum of Number of Records for Native-Country.

### 7) Capital-Gain



Sum of Number of Records for each Capital-Gain (bin).

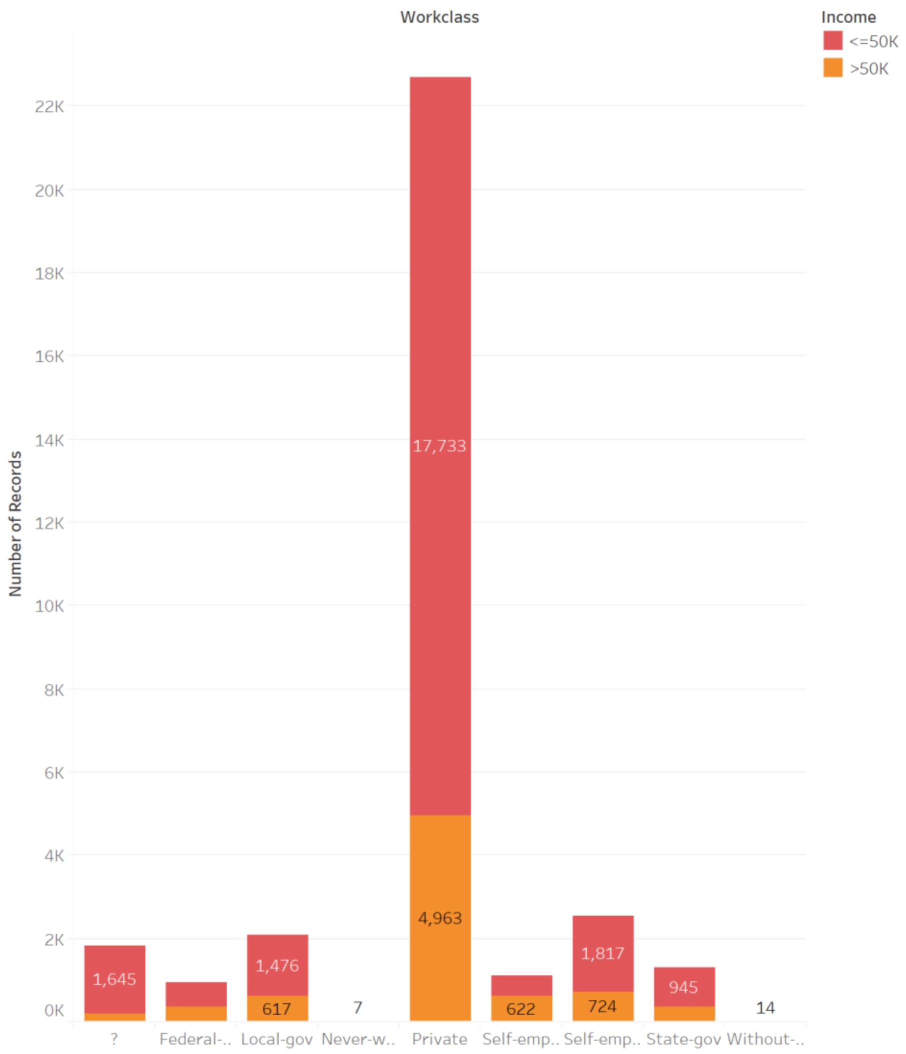
### 8) Capital-Loss



Sum of Number of Records for each Capital-Loss (bin).

## 9) Workclass vs Income

occupation_workclass



Sum of Number of Records for each Workclass. Color shows details about Income. The marks are labeled by sum of Number of Records.

**Observations From Tableau:**

- Only discrete values have missing values, and have '?' in case of null values
- Data is not evenly distributed as the number of observations of income '>50K'(~7K) is much less than the count of  those with income '<=50K'(~24K).
- Column 'education-num' and 'education' are two columns representing the same data, only in different data types.
- Capital-Gain and Capital-Loss have an extremely uneven distributions with more than 30K records having both these columns initiated to 0.

## Data Imputation

From the observations above, only the following columns have missing values in them

- 'workclass'
- 'occupation'
- 'native-country'

Missing values per column:

| | |
|---|---|
| age | 0 |
| workclass | 1836 |
| fnlwgt | 0 |
| education | 0 |
| education-num | 0 |
| marital-status | 0 |
| occupation | 1843 |
| relationship | 0 |
| race | 0 |
| sex | 0 |
| capital-gain | 0 |
| capital-loss | 0 |
| hours-per-week | 0 |
| native-country | 583 |
| income | 0 |

Since all the columns with missing values are discrete variables, we can make use of the following 2 methods to handle them:

1) Replace missing value with mean/mode.
2) Remove the records if there is null value for any of the attribute.

## Implementation Approach:

- Made use of a dictionary, with key as the column name of the data set and the value as a pandas Series.
  Classes_with_probability_yes = {column_names : Series}
  This Series has the probability of all the unique classes of each column given that the income = ”>50K”. The index of this series is the name/interval of the unique class.
  Similar dictionary is created for classes_with_probability_no.

- K-Fold cross validation is implemented using K = 10
  Data is shuffled before forming K-folds.
- These folds are then passed to predict() method where the test set is used for evaluation of the algorithm.

## Evaluation & Analysis

After implementing the code and after using K-Fold cross validation, following are the evaluation results.

| **Changes** | **Avg. accuracy** | **Avg. precision** | **Avg. Recall** | **Avg F1 Measure** | **Execution Time in sec** |
|---|---|---|---|---|---|
| - All columns considered<br>- No. of bins = 8 for all continuous variables | 81.78% | 0.5937767739 | 0.765749818 | 0.6687902222 | 89.88757992 |
| - Removed 'fnlwgt','education-num'<br>- Bin size = 5 for 'age'<br>- Bin size = 10 for 'hours-per-week' | 80.97% | 0.5794379986 | 0.7774241213 | 0.6638424385 | 74.91934037 |
| - Removed 'fnlwgt','education-num'<br>- No. of bins = 8 for all continuous variables | 80.48% | 0.5679247792 | 0.7861518033 | 0.6593377642 | 78.82492542 |
| - Removed 'education-num' column<br>- Bin size = 5 for 'age'<br>- Bin size = 10 for 'hours-per-week'<br>- No. of bins = 3 for 'capital-gain'<br>- No. of bins = 3 for ‘capital-loss’ | 80.38% | 0.5676485947 | 0.7704455997 | 0.6535653865 | 76.78704524 |
| - Using Gaussian Distribution for Discretization of continuous variables. | 81.08% | 0.5924216852 | 0.7700638019 | 0.6694990293 | 77.61646533 |