CSC869: Data Mining, San Francisco State University
Mini Project #1: Know Your Data, Naive Bayesian Classifier, and k-fold cross validation

1. **Due** by 11:55PM, Tuesday, March 12, 2019.

2. **Grading guidelines:** Posted in the top panel on iLearn.

3. **Dataset:** The dataset you will use for this project is the Census Income or Adult dataset, which is available at http://archive.ics.uci.edu/ml/datasets/Adult (Copy and paste this URL to your browser if clicking this link does not lead you to the webpage.) For a brief description of this dataset, click the Data Set Description link. To download the data, click the Data Folder link, then click the adult.data file to download the data. This dataset contains both categorical and continuous attributes. In addition, this dataset also contains missing attribute values.

4. **Problems**
   (i) Use basic visualization techniques to gain an initial understanding of the dataset. Specifically, you are required to visualize the relationship between each attribute and the class label. For a continuous attribute, you might need to discretize it first using a simple strategy such as equi-width. Please experiment with at least three different bin widths if you decide to discretize a continuous attribute. Observe these basic visualizations and summarize your main insights. You are strongly recommended to use Tableau for this task.
   (ii) Handling missing values: suggest and implement at least two strategies to handle the missing values for categorical and numeric attributes, respectively. These strategies should be based on your observations made in the previous step.
   (iii) Implement a Naïve Bayesian Classifier for this dataset. This dataset contains continuous attributes. Take the following two different approaches to handle a continuous attribute: (1) using the equal-width binning method to transform this attribute into a categorical attribute before building the classifier. Select a "proper" width based on your observations made in step (i); (2) assume this attribute follows a Gaussian distribution.
   (iv) Implement the k-fold cross validation strategy and evaluate your classifier by setting k=10. Also evaluate the impact of different strategies implemented in step (ii) for missing values and the two approaches for handling continuous attributes in step (iii).

5. **Requirements**
   (i) Individual work only.
   (ii) You need to implement this classifier in one of the following programming languages: C, C++, Java, or Python.
   (iii) To validate your implementation, you can compare yours with an existing implementation, for instance, the Naive Bayisian Classifier included in the Weka data mining suite.
   (iv) Use 10-fold cross validation to evaluate your algorithm. Adopt the classification accuracy, i.e., #(records correctly classified in the test set)/#(total records in the test set), precision, recall and F1-measure to measure the quality of your classifier. Implement this evaluation module either as a separate program or a subroutine in the program implemented in (ii).

6. **Submission instructions**
   (i) Archive the following items into one compressed file with your name in the file name, e.g., JohnDavis-NaiveBayesian.zip:
      a. Source code with comments in the language of your choice.
      b. Instructions on compiling and running your program.
      c. A brief description of the main steps that you have adopted in accomplishing this project. For instance, have you done any data preprocessing tasks? If yes, what are they and why are they necessary?
      d. Evaluation strategies, results and a brief discussion. For instance, are the results acceptable? What can you do to improve the results?
   (ii) Submit the above file on iLearn. No late submission or e-mail submission will be accepted.
7. **Project demonstration**

You will be required to demonstrate your work in class or during the instructor's office hours. You will be asked to explain the major functions in your program(s) as part of the demonstration.