# CSC-272: Big Data Spring 2020

**PROBLEM SET #2 – Due Wednesday, February 5 @ 10:30am**

**Name: Boone Tison**

*All submissions are to be completed using Word or another word-processor.*

1. Use a text editor (e.g., VSCode, Notepad, etc.) to open at least 3 of the .arff files provided in the Weka data folder. Examine the contents. Review the section on the ARFF format in the text and answer these questions.

1. Excluding comments (which begin with a % sign), what are the three sections of an ARFF file, and what keywords designate each section?

@relation – the name of the file

@attribute – lists the attributes

@data – the instances

1. Describe how both *nominal* and *numeric* attributes are designated.

Numeric attributes are designated by the keyword numeric. Nominal attributes use brackets to designate the available labels.

2. To explore the details of the ARFF format and the process of data pre-processing, we’ll next build an input file for Weka. Visit <http://archive.ics.uci.edu/ml/datasets/Adult> and download the “Census Income” dataset. This involves downloading two files – adult.data and adult.names. (For future use, download adult.test as well.) These files are in a simple comma-delimited format, so they will have to be converted into a single ARFF file for Weka. Open both files in a text editor and examine them. Create and save a new file with the .arff extension and copy all of the data from adult.data into it. Then use the information from the adult.names file to manually construct a file meeting the ARFF format that we’ve studied. Briefly describe the steps that you took to generate the ARFF file below, including any errors that you had to correct. (*Hint*: Remember that the order of listed attributes must match the order of the data itself, and by default the class attribute should be listed last.)

After copying the data, I created, at the top, the relation using the name us census. I then created the attributes using the names in the names file. If the file said the attribute was continuous, I made it numeric. If not, I used the labels in the curly brackets.

3. Understanding the dataset is a key step in data mining. Open your new file in Weka and use the **Edit…** button to study your data. Make sure the income level attribute is set as the class attribute. (It will be automatically if it is listed last in the dataset.) Write down two issues or hypotheses that might be interesting to investigate. (For example: You could switch the class attribute to “sex” and try to predict that variable from the rest of the variables.) You can also consider other issues, such as preprocessing the data to possibly improve predictive performance. Explain what made you think of each issue/hypothesis, and how you think data mining might be used to find an answer.

For clarity, state each hypothesis as a question. (For example: “Can a person’s sex be predicted by the other attributes?” or “What attributes are most predictive of a person’s sex?”) Don’t just list two variations of this hypothesis. Be creative.

Can sex be used to predict occupation? Does native country correlate with lower income?

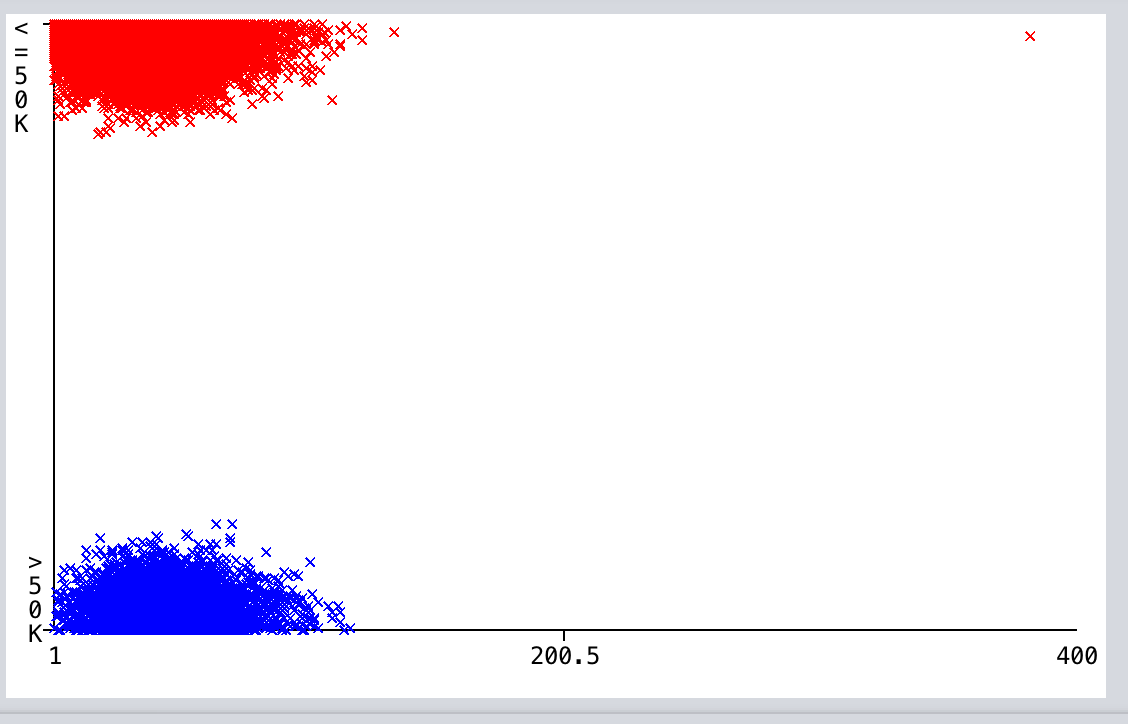
4. Give one example of a nominal attribute in the data set that is also *ordinal*. And one example of a nominal attribute that is not ordinal. Explain.

Sex is a nominal attribute that is also ordinal. Sex could be set to 0 or 1, male of female. Native country is a nominal attribute that is not ordinal, because there are too many options to make it ordinal.

5. Give two examples of missing data in the data set, and for each give a *different* plausible explanation for why a value is missing. Then give an example of a missing value that may have significance to the learning algorithm (i.e., its absence might enable something to be learned rather than being meaningless), and explain what that significance might be.

Instance 28 is missing occupation, this could be because the instance’s occupation status is unclear at the moment. Instance 15 is missing native country, this could be because the instance’s native country was not listed as an option. There are several instances missing their work status. These could help with the class attribute.

6. In the first instance (first row of the dataset), edit the hours-per-week attribute to be 400 instead of 40. Then go to the **Visualize** tab. Display a graph of hour-per-week relative to income level. Use the jitter slider to visualize the data better. Make a screen capture of the graph and paste it below. What can you learn from this graph, pertaining to the value that you changed? (Make sure to re-edit the data back to its original state after this step.)



This instance is now a significant statistical outlier. Most instances have a much lower number of hours.

7. Still in the Weka preprocess tab, look at how each feature correlates with the class attribute using the histogram in the lower right-hand corner. (You can look at all of them at once by clicking on **Visualize All**.) List two attributes that you think might be useful in predicting the class attribute, and – in a sentence or two – justify why the visualizations make you think these attributes will be useful.

Marital status could be useful, married with civilian spouses’ instances have a significant proportion of people who make over 50k. Education shows that instances with higher education have better likelihood to make more money.

8. Run Weka’s J48 classifier on the dataset with the test option set to a percentage split of 66%. We haven’t used the percentage split option yet. It’s very simple – it uses only 66% of the data instances to learn, and the remaining instances to test. Visualize the decision tree. Answer the following questions:

1. What is the accuracy of the classifier on the test data?

The accuracy is 86.5324 %.

1. How many leaves are there in the tree? (Hint: You don’t actually have to count them!)

There are 564 leaves on the tree.

1. How long did it take to build the model?

The model was built in 1.24 seconds.

9. Now run the ZeroR classifier from the rules set, using the 66% percentage split for testing.

1. What is the accuracy of the classifier on the test data? How long did it take to build the model?

The accuracy is 76.5062 %. It took 0.14 seconds to build the model.

1. Do some research (from the textbook or online) to figure out how this algorithm works, and briefly explain.

ZeroR predicts the majority class. It is useful to obtain a baseline.

1. In light of your explanation, give a thorough interpretation of the accuracy measure reported in part (a).

The data’s class attribute, the income level, is 76.5062 % of the time <50k. There is a large majority for the class attribute, however this does not mean it will be predictive.

10. Are you surprised by the accuracy results obtained in questions 8 and 9? Whether you are surprised or not, briefly articulate a lesson learned, or an insightful observation from a comparison of these results.

J48’s accuracy did not surprise me. Setting aside some testing data should improve the accuracy. Getting 86% on such a large dataset with many attributes and labels seems like a good result. The ZeroR’s accuracy was also not surprising knowing the large majority in the class attribute. I learned the value of the percentage split testing method.

This is a good source for background reading on questions 8-10: *https://machinelearningmastery.com/estimate-baseline-performance-machine-learning-models-weka/*