

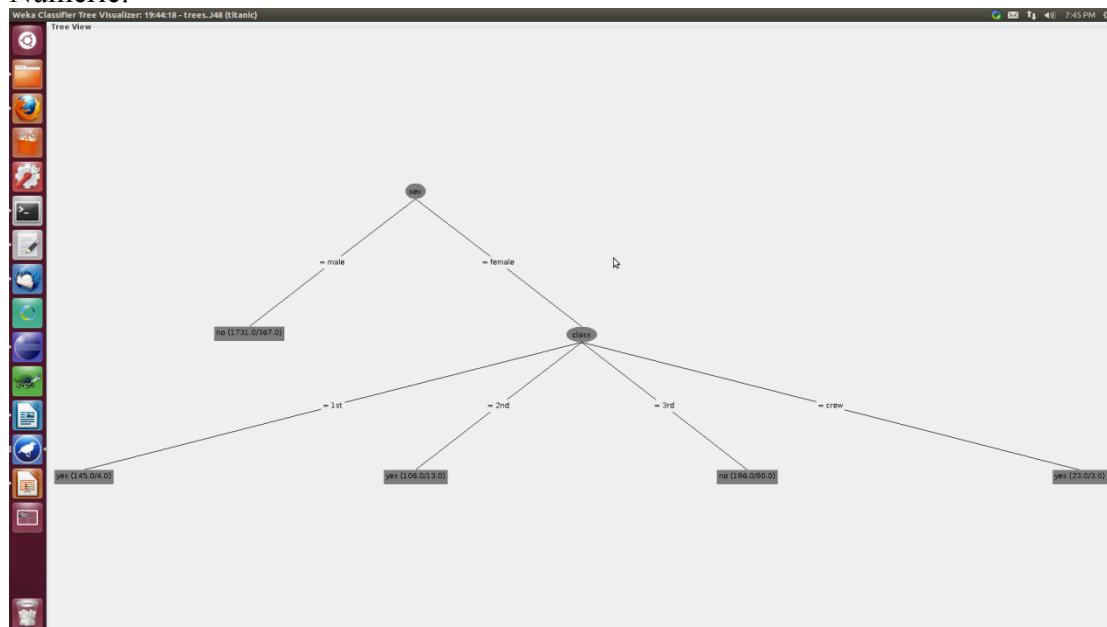
Assignment 2 Answer Key

Part A:

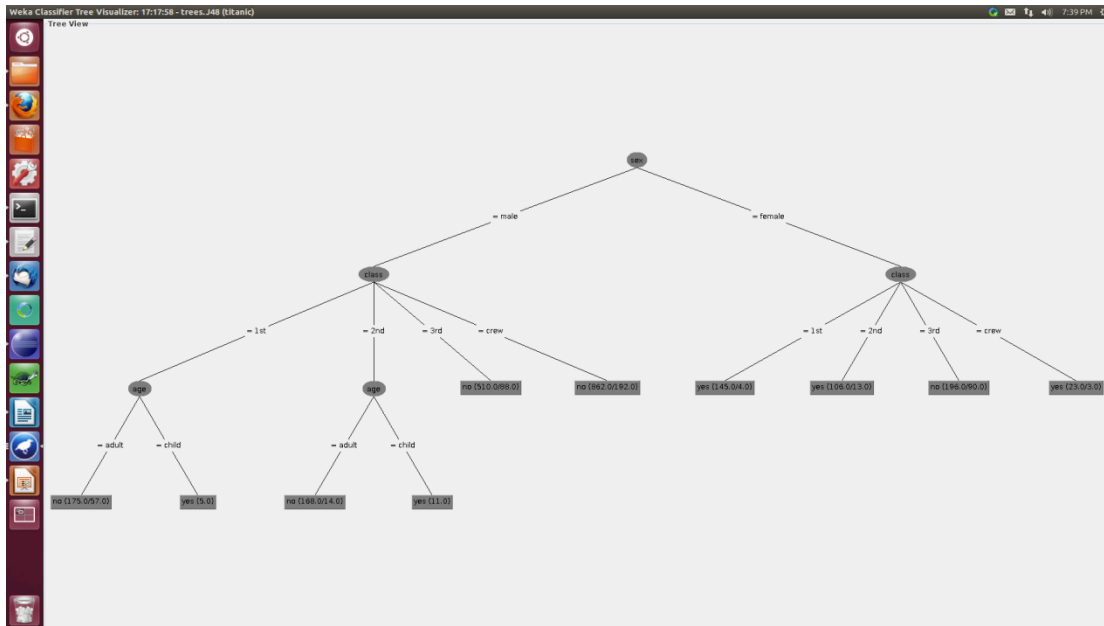
For this part you will use the three versions of “titanic.arff” that are given in the Assignment 2 folder. Basically this data contains information about the passengers in the titanic. The attributes are crew, age, sex and survived (or not). In the original titanic.arff file, age is represented as a nominal attribute with values of child or adult. In titanic-age-numeric.arff, age is represented instead as a numeric attribute representing the actual age of the passenger. The final file “titanic_noise.arff” is a version of “titanic.arff” with a little bit of noise added to one of the attributes in two instances.

1. Using Weka, Visualize/Draw the tree generated for the three sets of data. In order to do this for one file, load it on the Preprocess tab, then go to the classify tab. Choose J48 as the learner. Run the experiment. Right click in the results buffer to generate the tree. Paste an image of your tree in your assignment write up.

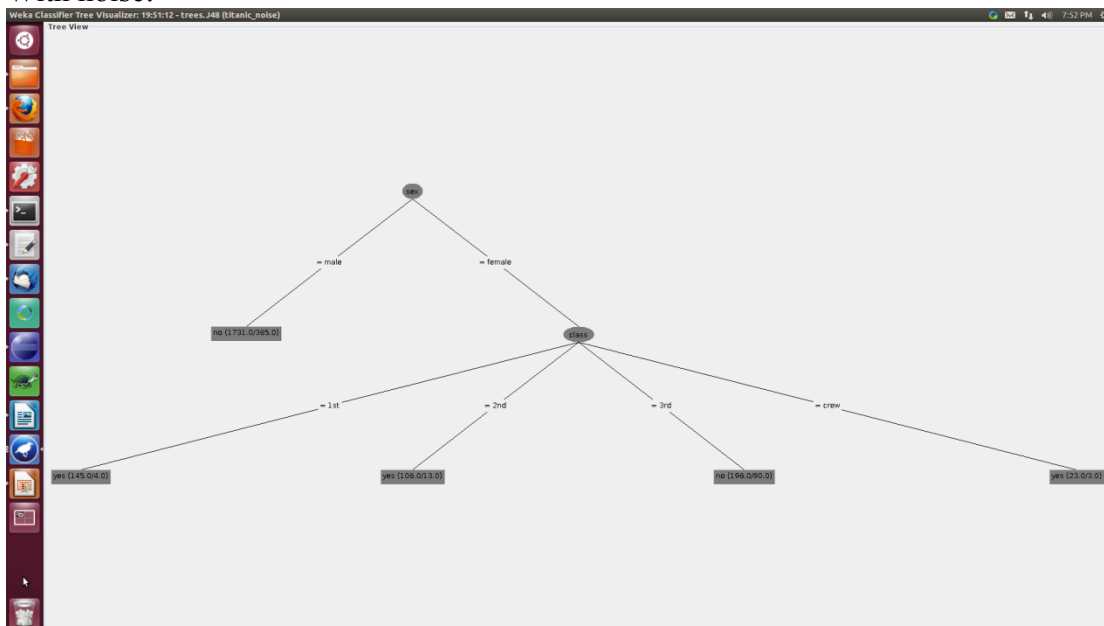
Numeric:



Nominal age:



With noise:



- Now compare the results from the original file where age was a nominal attribute with the results where age was a numeric attribute. How are the results different? How is the tree different? Considering that it would be possible for the exact same tree to be learned (because a split at age 18 would produce the same child/adult distinction), why do you think J48 settled on a different model?

By replacing the binary age attribute with a numeric one, the attribute more finely differentiated between instances. This introduced more uncertainty into the decision making about how to use the attribute since now the split can occur in many different places rather than one. So it seems more risky to split on that attribute. J48 first builds out the tree and then prunes off the risky seeming branches. Since the age attribute seems more risky when encoded as a number, and since the dataset is relatively small, it must not have considered it worth the risk to split on age on the male tree when using this new feature encoding.

3. Now compare the results from the original file with the file that has noise added. Describe the effect of noisy data on the generated machine learning model. Based on the differences between models, which kinds of instances do you think the model would consistently make mistakes on?

Nominal: Kappa: 0.429, accuracy 78.92

Nominal one with noise: Kappa: 0.41, accuracy: 78.4

The noise caused the tree to collapse in the male branch. The reasoning is similar to that in the previous question. The noise made the split seem less certain. Thus, it seemed more risky, and the split was then removed at the pruning stage.

14 instances were classified differently using the noisy model (this is the number of additional incorrectly classified instances between the original model and the noisy model).

J48's performance on the 'no' class is the same for the two data sets, i.e., the noise only caused the prediction on 'yes' class to change. The noise caused the 'male' branch of the tree to be pruned, which increased the number of correctly predicted 'yes' instances.

4. Turn in your answers to questions 1-3

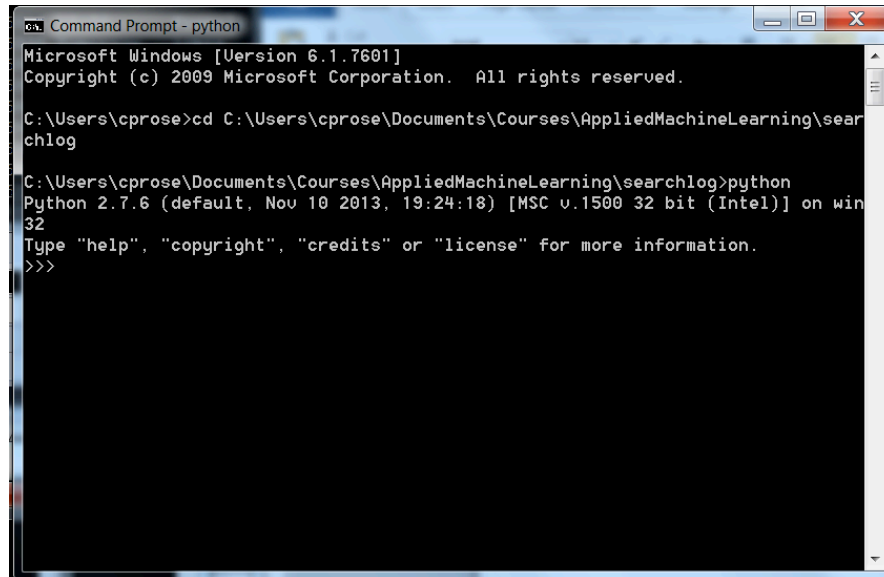
See above.

Part B:

Here is a step by step guide:

1. First, if you do not have a version of Python running on your computer you will need to install it. I suggest you install version 2.7. You can get to the download page here: <http://www.python.org/download/releases/2.7/>. If you are using Windows or Mac OS, I suggest you use one of the Windows or Mac OS installers that is appropriate for your operating system. Feel free to ask the TAs if you need help installing the software. When the code is installed properly, you should be able to open a command line prompt, type python, and launch the runtime environment. First cd into the directory you created from the zip file and then launch python. Note that I have included some Python cheat sheets in the assignment 2 folder. You

can also search for the answer to questions on google easily, especially the stackoverflow website. That is a great resource to get used to using.



```
Command Prompt - python
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\cprose>cd C:\Users\cprose\Documents\Courses\AppliedMachineLearning\searchlog

C:\Users\cprose\Documents\Courses\AppliedMachineLearning\searchlog>python
Python 2.7.6 (default, Nov 10 2013, 19:24:18) [MSC v.1500 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

2. Now run the python script you were given using the command `execfile('SetupSearchData.py')` at the python prompt. It will take a few moments to run, but then you should have a file called `output.csv` in the directory. This output file could be read into weka to run experiments. You can feel free to experiment with that, but it is not required for this assignment. You can compare your file with `outputReference.csv` to make sure you created the file properly.
3. Now what you are required to do is to modify the script to add an additional variable. This variable will be analogous to `WikiAccess`, but instead of tabulating the number of accesses to Wikipedia, it will tabulate the number of accesses to google related pages (i.e., web pages with google in the URL). Once you have made the modifications, run the script to create the updated `output.csv` file. What you will turn in is your modified script and your new `output.csv` file.

See additional files.